Machine Learning-Based Cellular Traffic Prediction Using Data Reduction Techniques

Cellular traffic prediction can help mobile networks organization to allocate sufficient resources to manage traffic but current traffic prediction algorithms are trained on huge available data which require heavy time complexity and resources. To overcome from this issue author of this paper employing a novel algorithm called AML-CTP (Adaptive Machine Learning-based Cellular Traffic Prediction) which will trained on small set of accurate dataset to reduce time complexity and to enhance prediction accuracy or R2SCORE.

To select accurate small training data author has used following technologies

1. First dataset will be clean and normalize using MIN-MAX SCALER algorithm. This step is called Phase1 Data Processing
2. Select-K-Best algorithm was applied on clean dataset to select top features from the dataset
3. PCA dimensionality reduction algorithm applied on selected features to reduce dimension by selecting un-correlated features. This step is called as Phase2 Data Reduction
4. DBSCAN and Kernel density Clustering algorithms: Applied density based DBSCAN clustering algorithm to group similar values into same cluster and then applied Kernel density to generate another cluster and then both DBSCAN and Kernel Density co-matrix similarity will be calculated to know how well points are arranged in both clusters. Clusters with high similarity will be selected to train with ML algorithms. This also comes under Phase2
5. Training Models: selected high similarity cluster will be trained with multiple ML algorithms such as SVM, Light Gradient Boosting, Decision Tree and Linear Regression. This step is called as Phase 3
6. Model Selection: best performing model will be selected and this process is known as Phase 4 model selection

In propose paper instead of training entire dataset author is training a cluster which has high similarity, so data will be distributed into multiple clusters which will reduced dataset size and training complexity time will also be reduced. For example if dataset has 10000 records and if distributed this into 3 clusters then each cluster will have approximate 3500 records so instead of training 10000 records propose paper will train only 3500 records. NOTE generated each cluster will have different size of records like 3000, 4000, 2000, 1000 etc.

Above ML algorithms performance will be evaluated in terms of R2Score (accuracy), RMSE (root mean square error) and MAPE (mean absolute error percentage. ML model with high R2score and less RMSE and MAPE will be consider as best performing model.

RMSE and MAPE refers to difference between predicted and true values so the lower the difference the better is the model. R2score refers as accuracy for the forecasted algorithm.

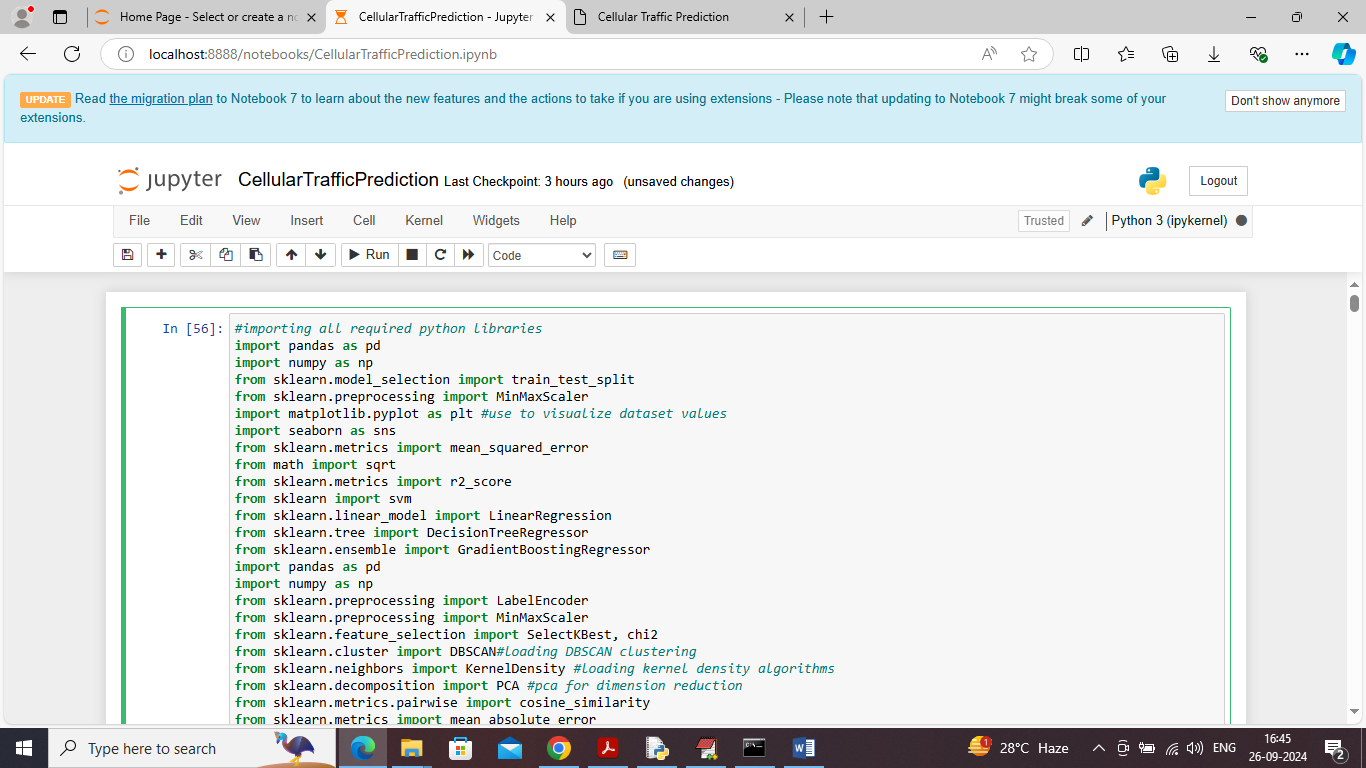
To train above algorithms author has used Cellular LTE dataset from Egyptian company but that dataset is not available on internet so we downloaded LTE cellular traffic dataset from IEEE paper which can be downloaded from below URL

<https://ieee-dataport.org/open-access/crawdad-srfglte-4g-highway-drive-tests-salzburg>

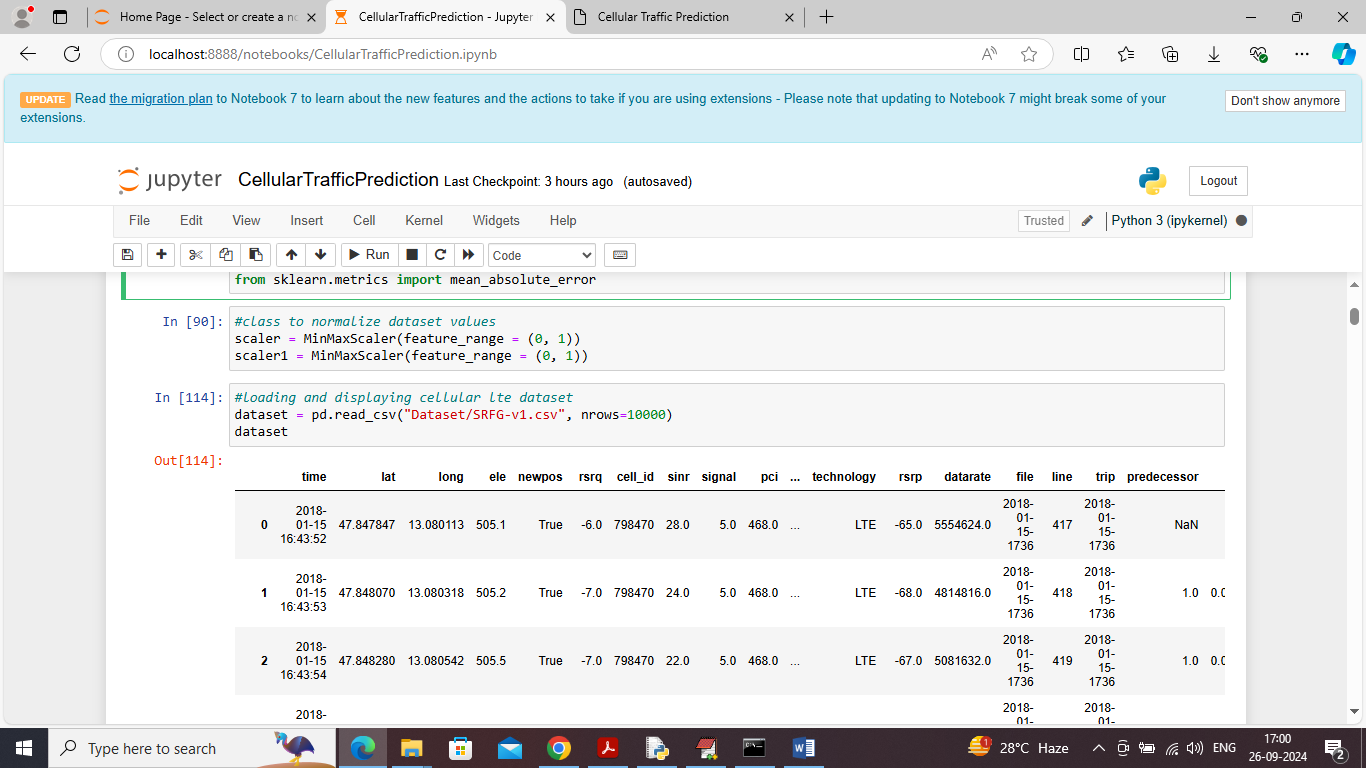
Extension Concept: in propose paper to select best model author has used all traditional ML algorithms but not utilized any advance ML algorithms like XGBOOST, CATBOOST or Voting Regression so as extension we have experimented with all 3 algorithms by tuning each algorithm parameters but in all algorithms tuned XGBOOST with 150 estimators giving best R2score compare to all propose algorithms.

SCREEN SHOTS

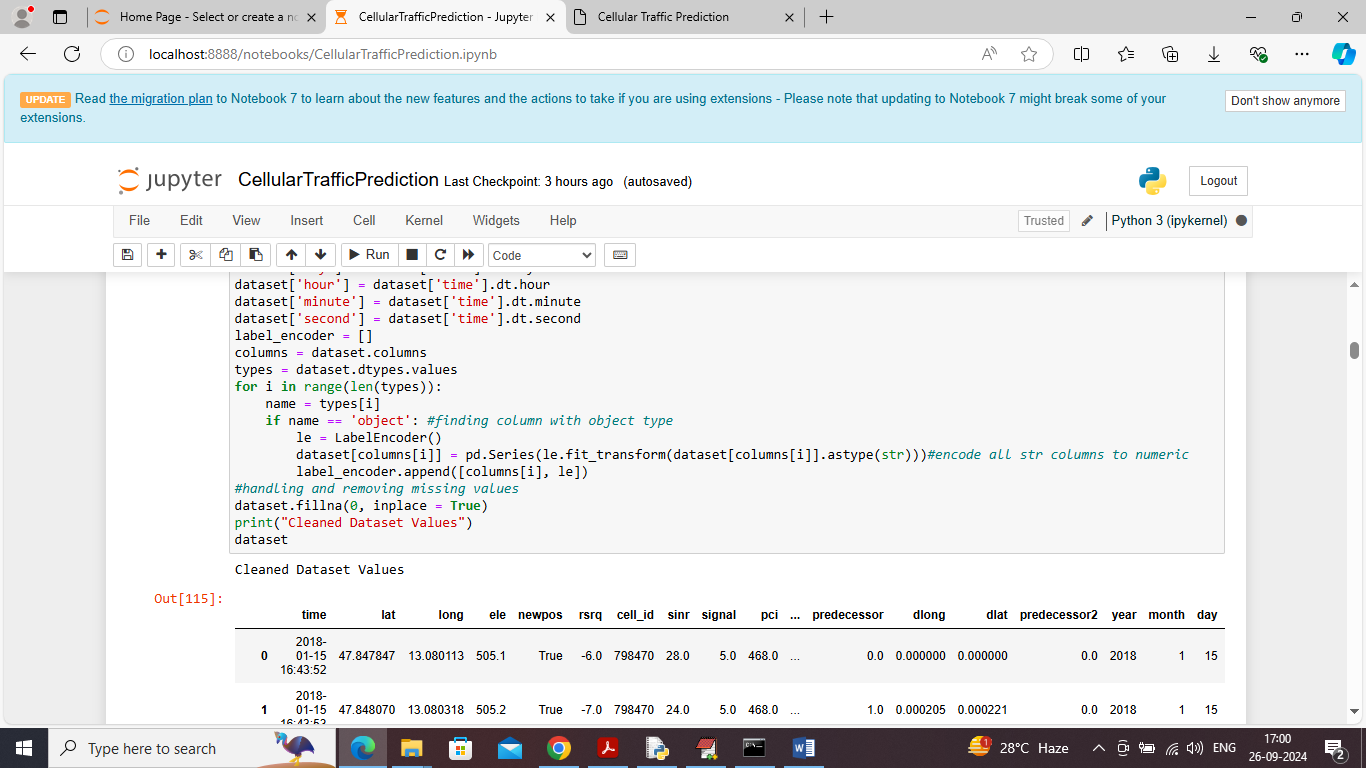
We have coded this project using JUPYTER notebook and below are the code and output screens with blue colour comments



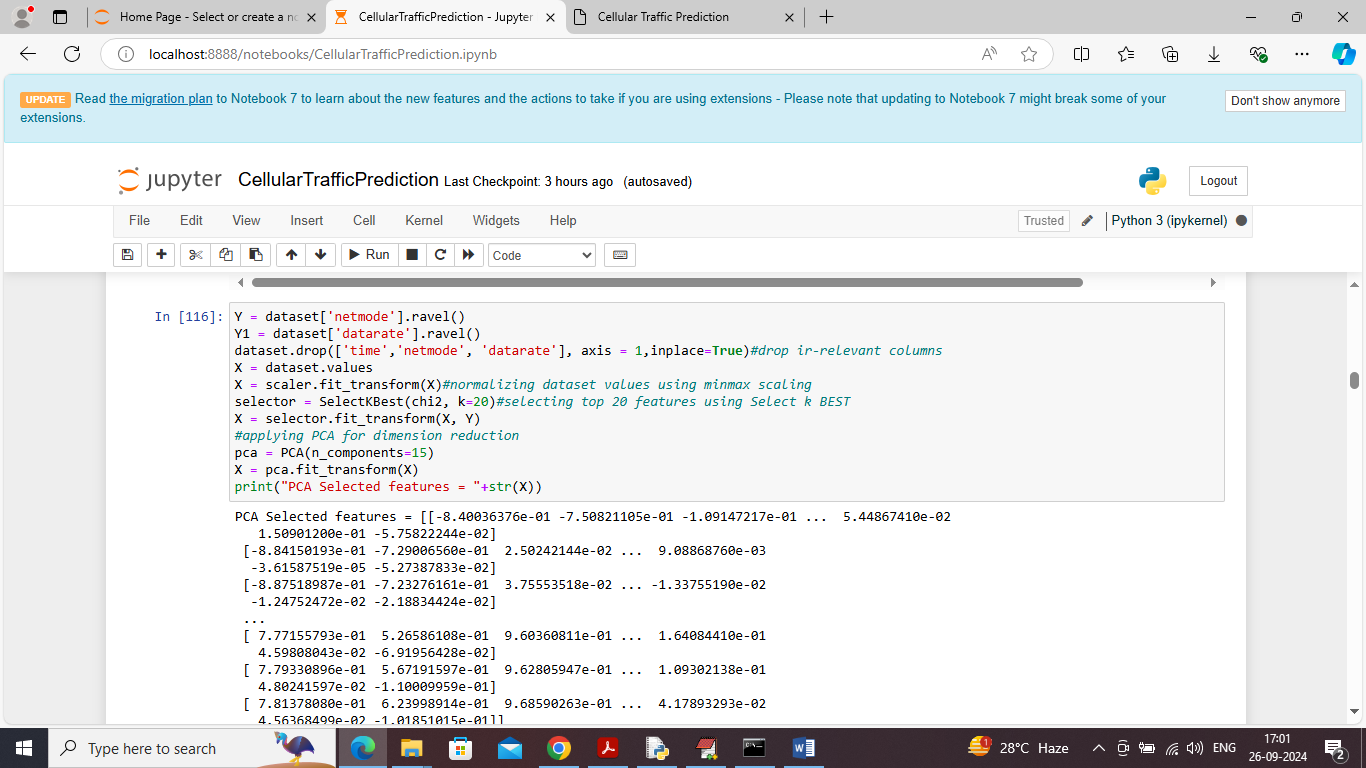
In above screen importing required python classes and packages



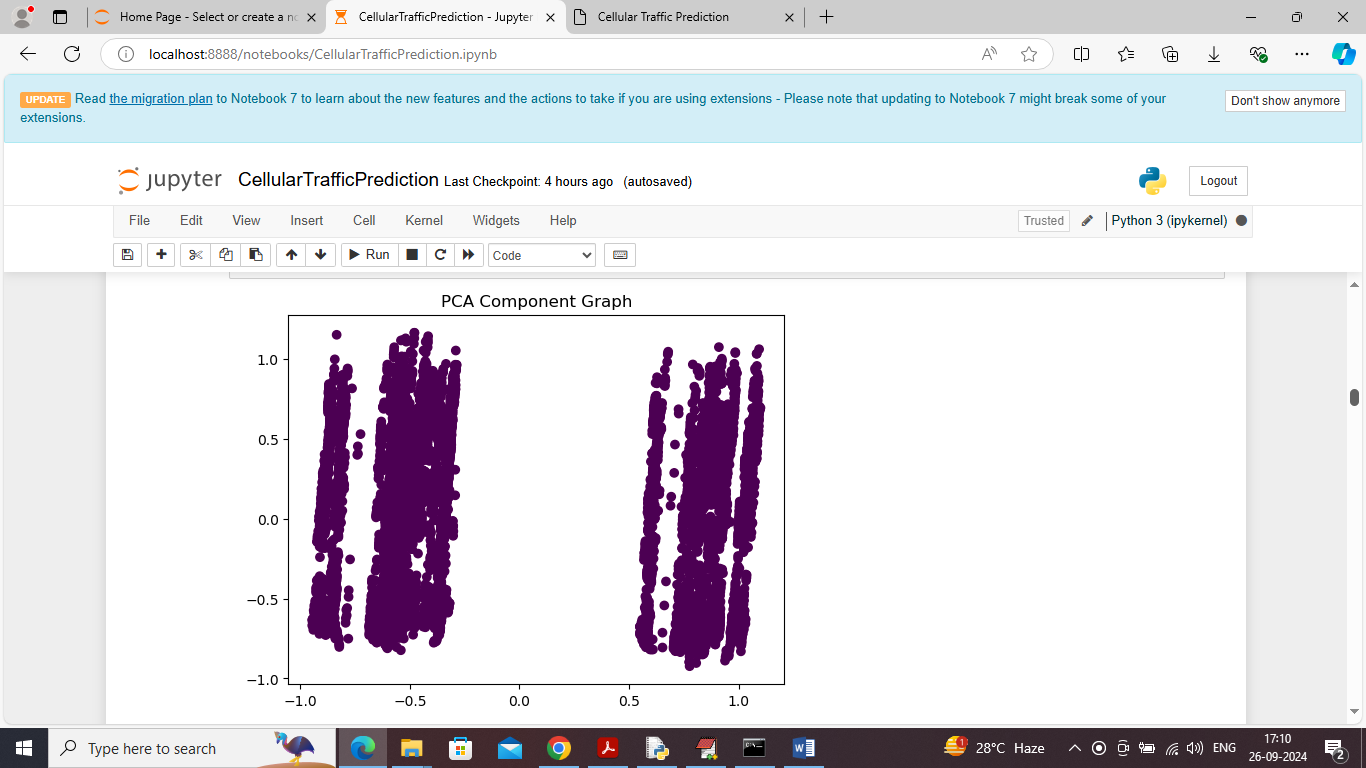
In above screen defining MINMAX scaling to normalize dataset values and then loading and displaying dataset



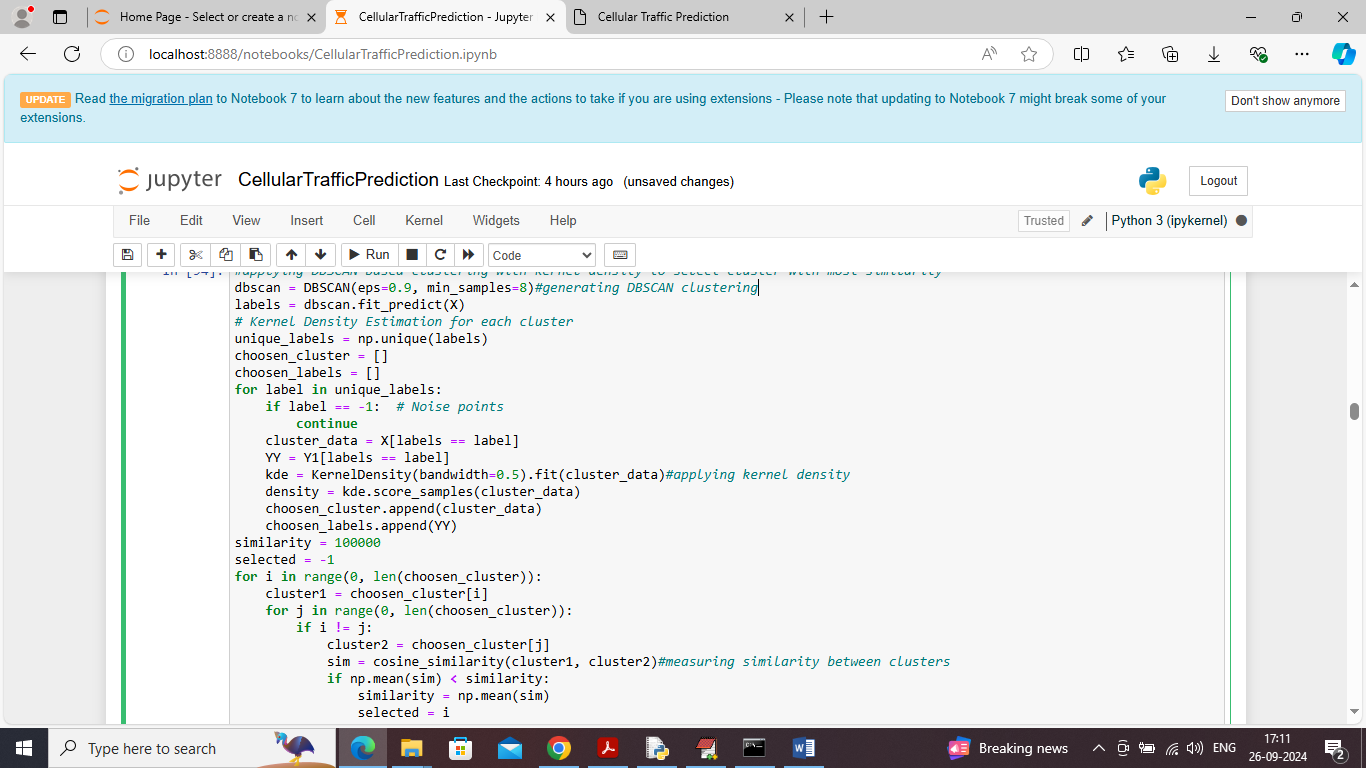
In above screen applying dataset processing technique to convert non-numeric values to numeric values and then replacing missing values with 0 and then can see all values are converted to numeric format



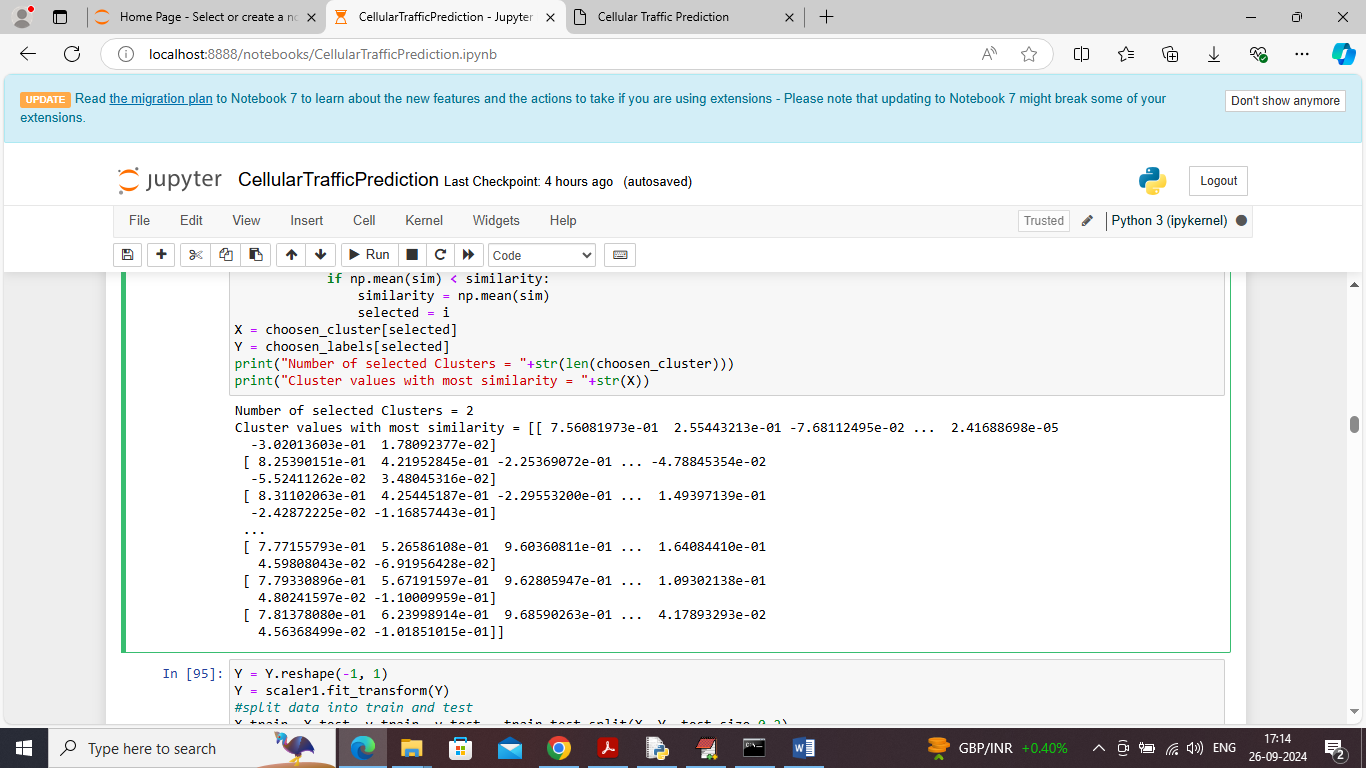
In above screen applying select K BEST algorithm to select top features and then employing PCA dimension reduction algorithm to reduce features and then displaying reduced features values



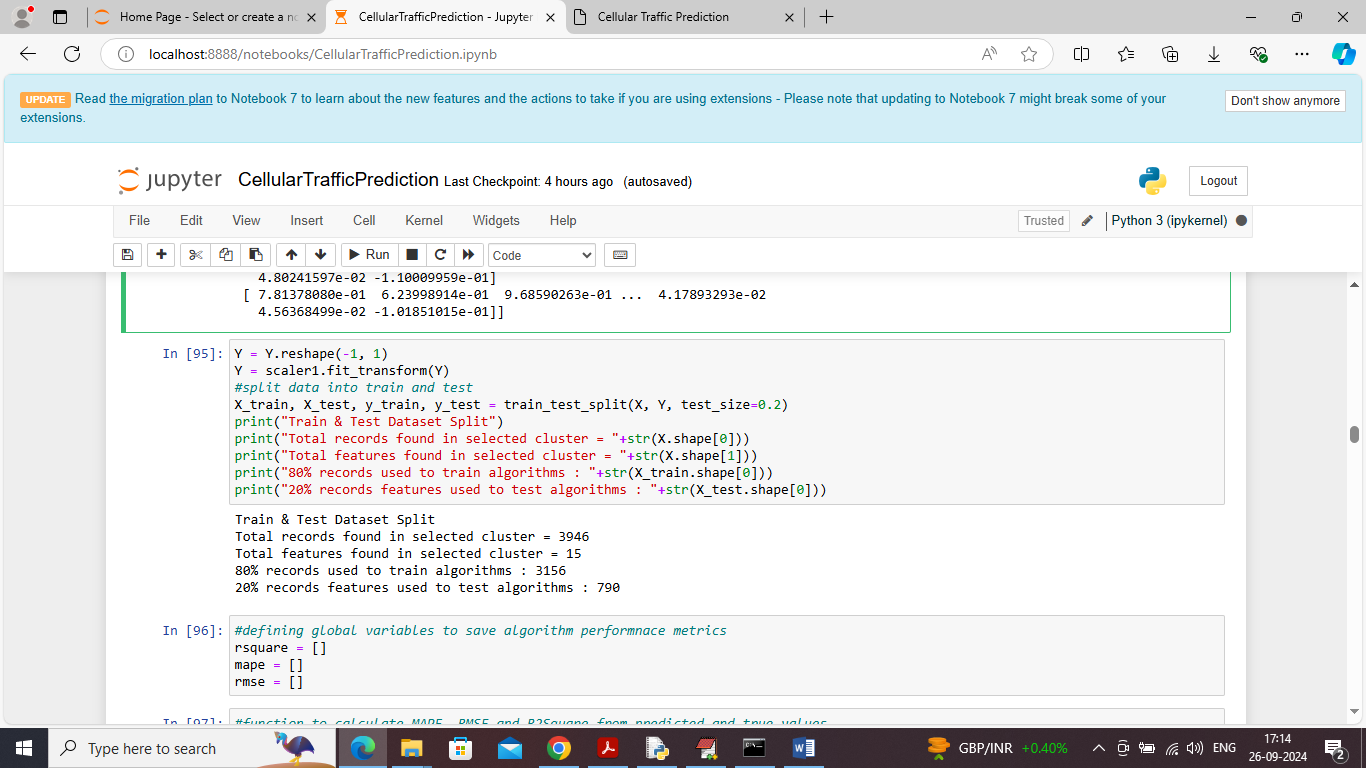
In above screen visualizing PCA features graph where can see all similar points are at one place and other similar points are in different place



In above screen applying DBSCAN, Kernel Density clustering to select most similar cluster by measuring similarity between both clusters and will get below output



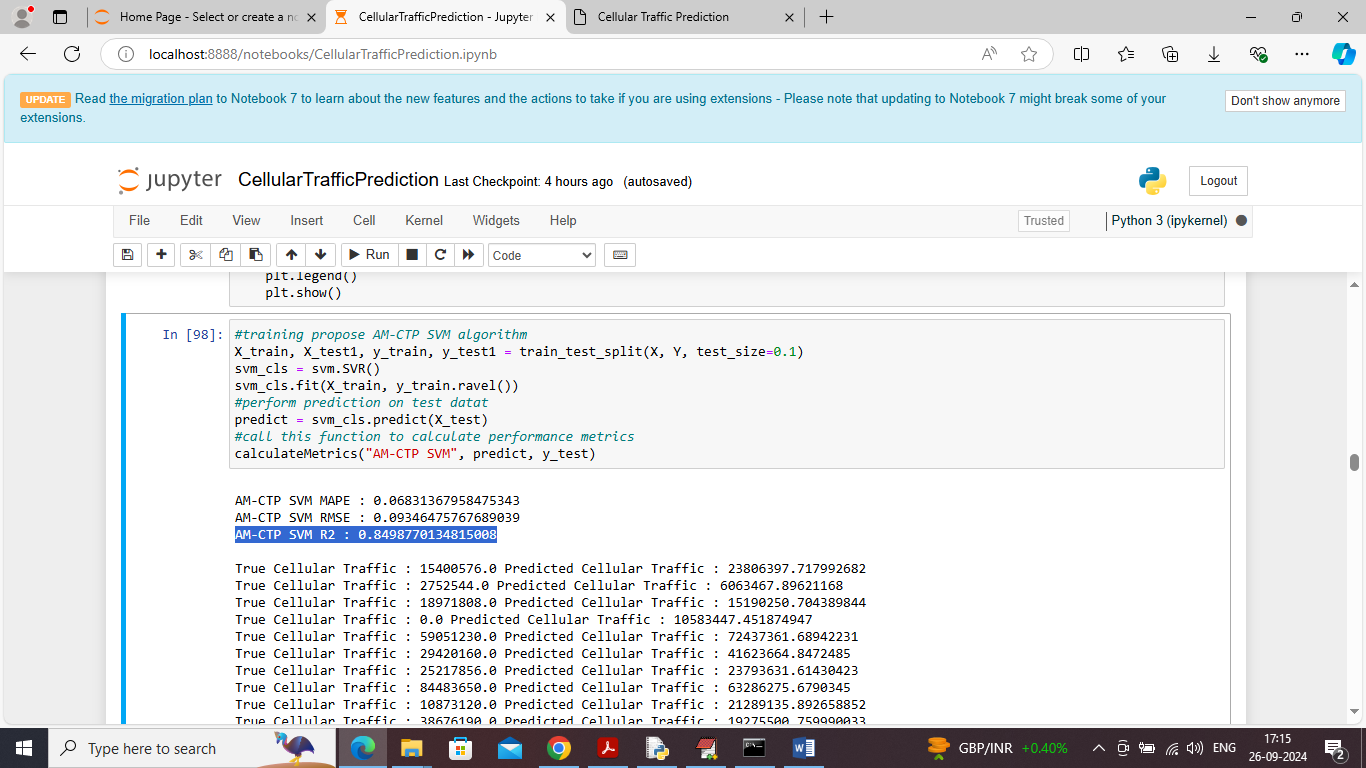
In above screen can see most similar selected cluster values



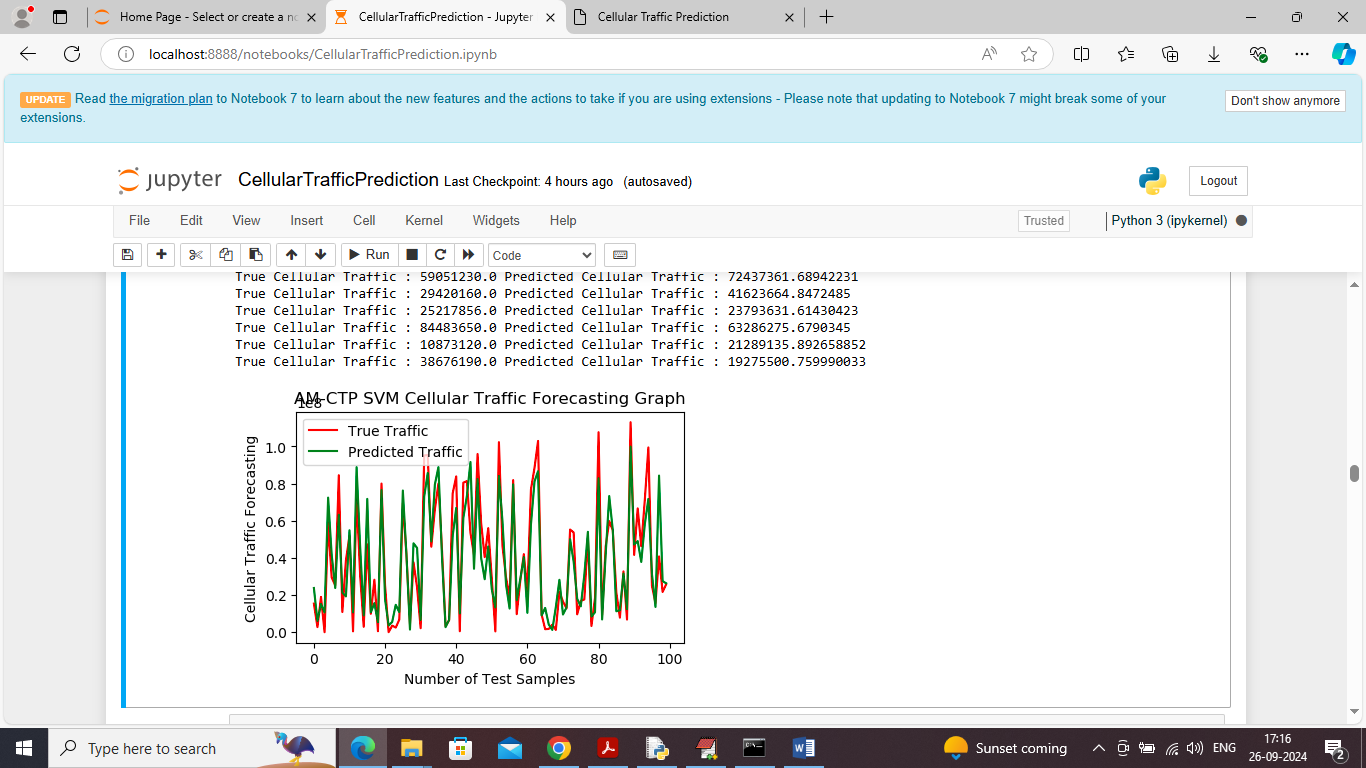
In above screen splitting dataset into train and test where application using 80% dataset for training and 20% for testing



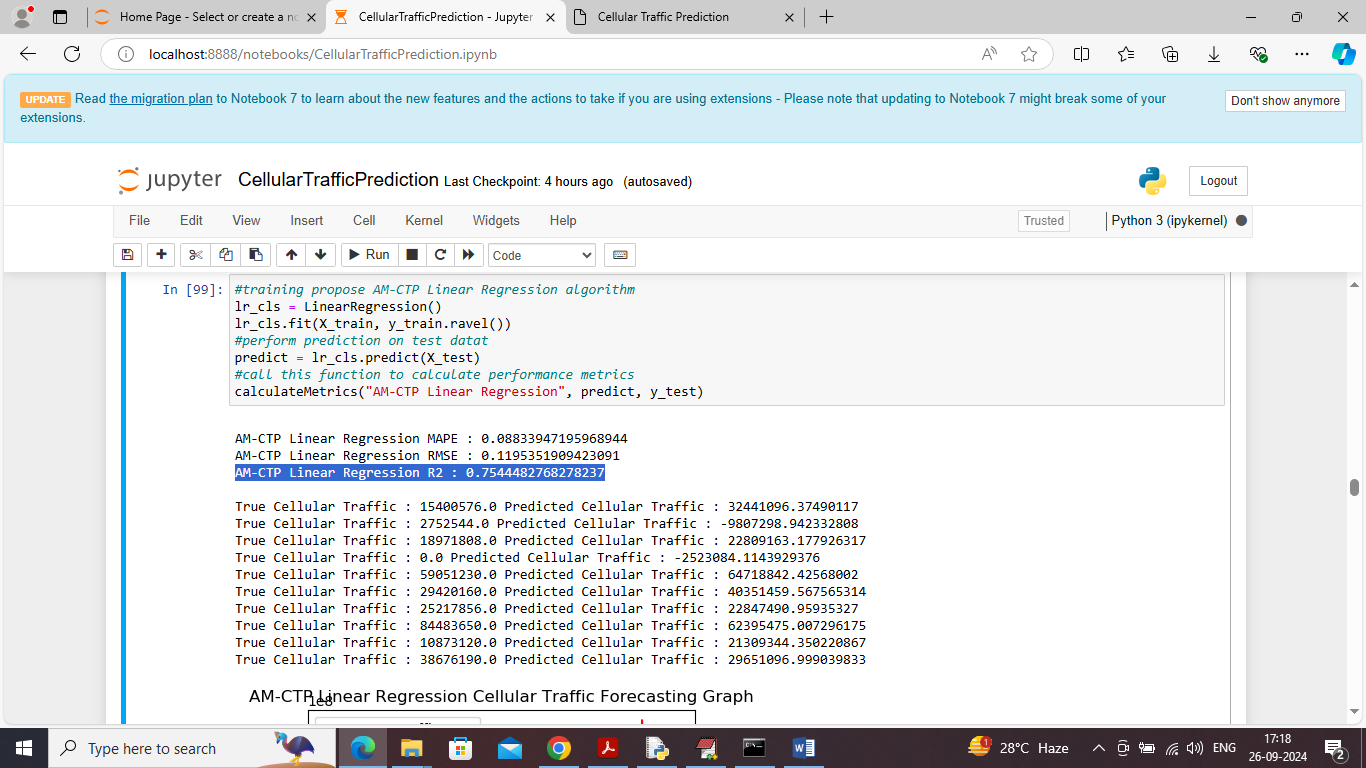
In above screen defining function to calculate R2score, RMSE and MAPE



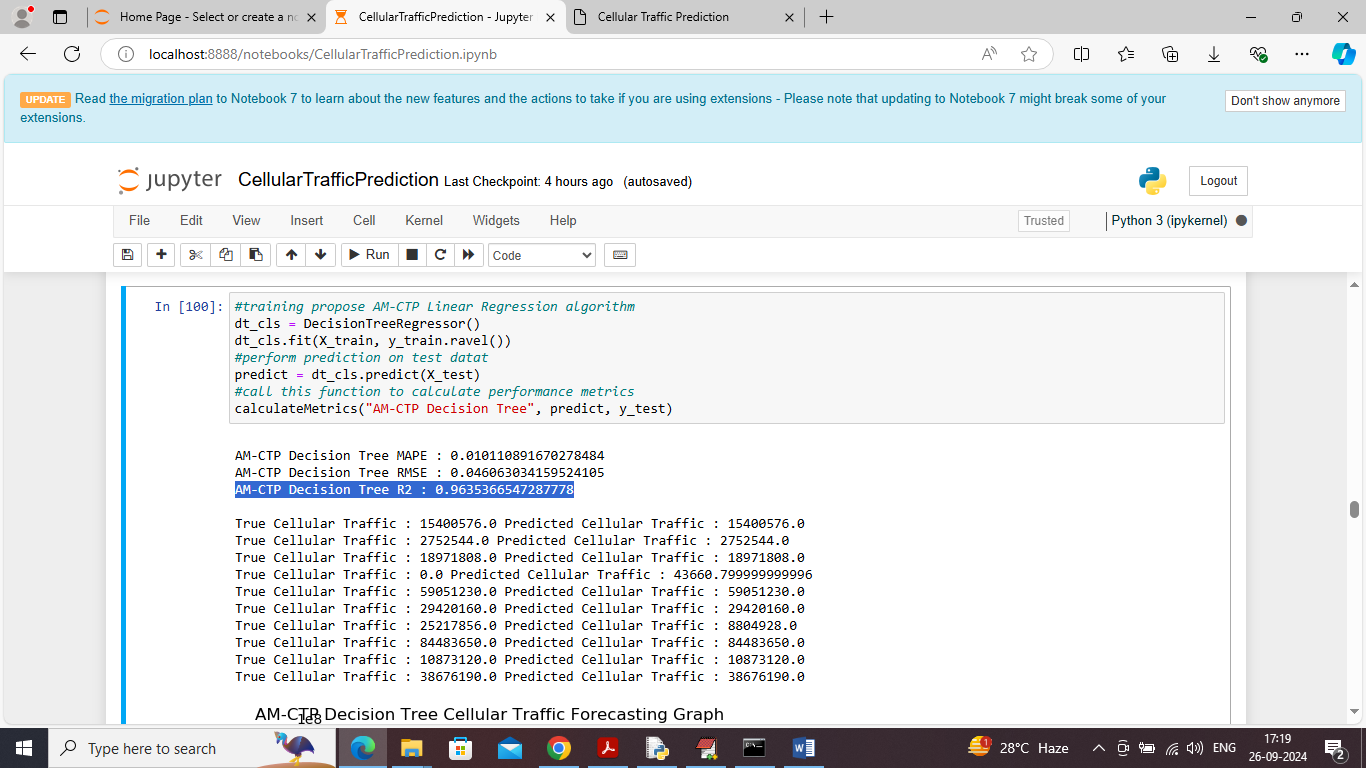
In above screen training propose SVM algorithm and then can see RMSE, MAPE error values and then can see R2 score as 84% and then in next line can see True cellular traffic and predicted traffic values



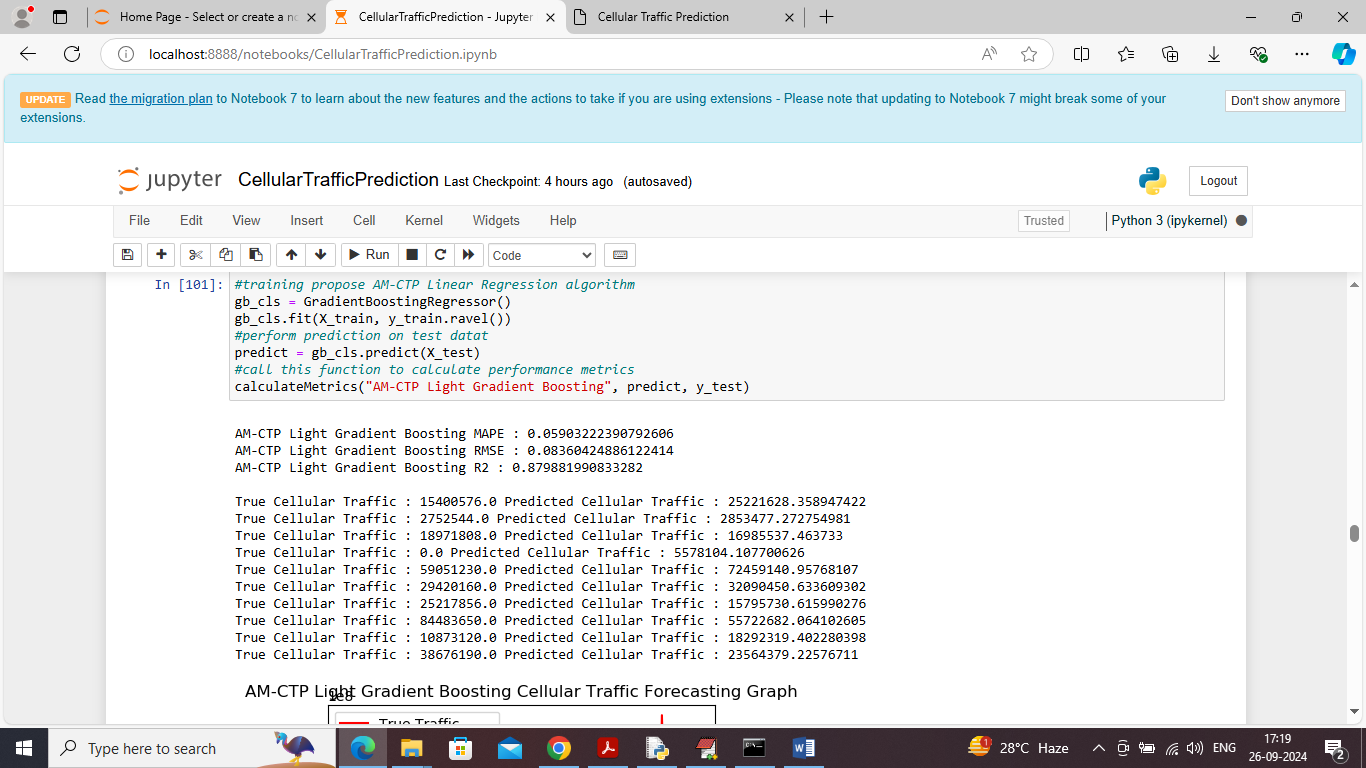
In above graph x-axis represents Number of Test samples and y-axis represents cellular traffic and then red line represents true traffic and green line represents predicted traffic and can see both lines are fully overlapping with little difference so both true and predicted values are close



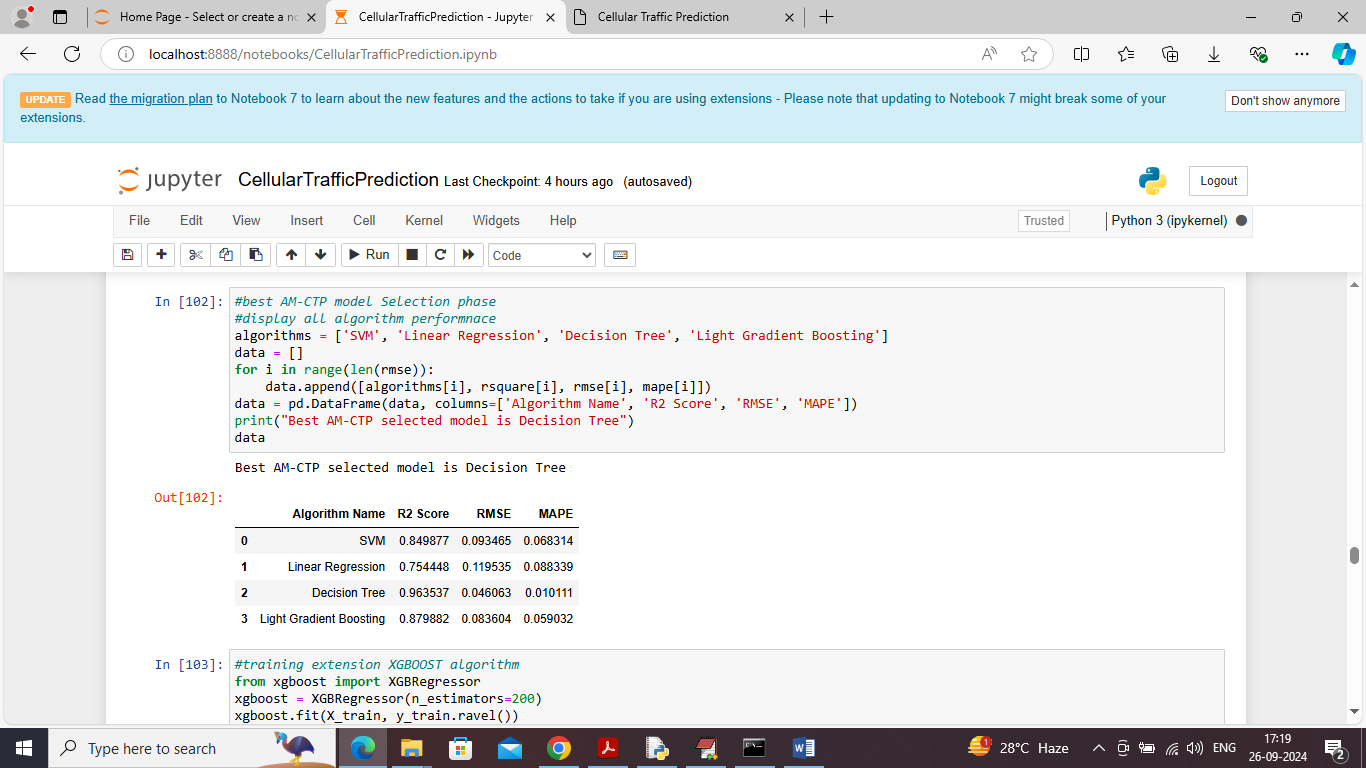
In above screen linear regression got 75% r2 score and can see other metrics with prediction graph



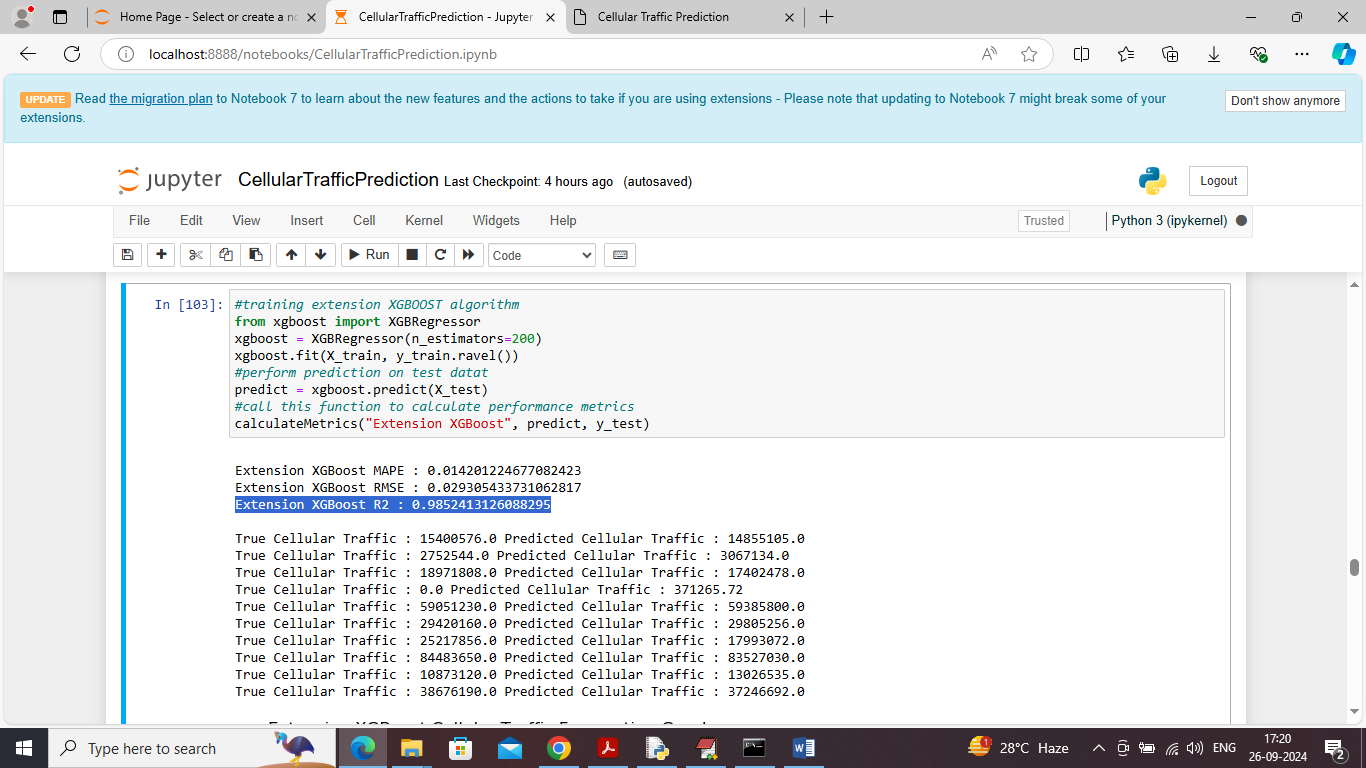
In above screen Decision tree got 96% r2 score and can see other metrics with prediction graph



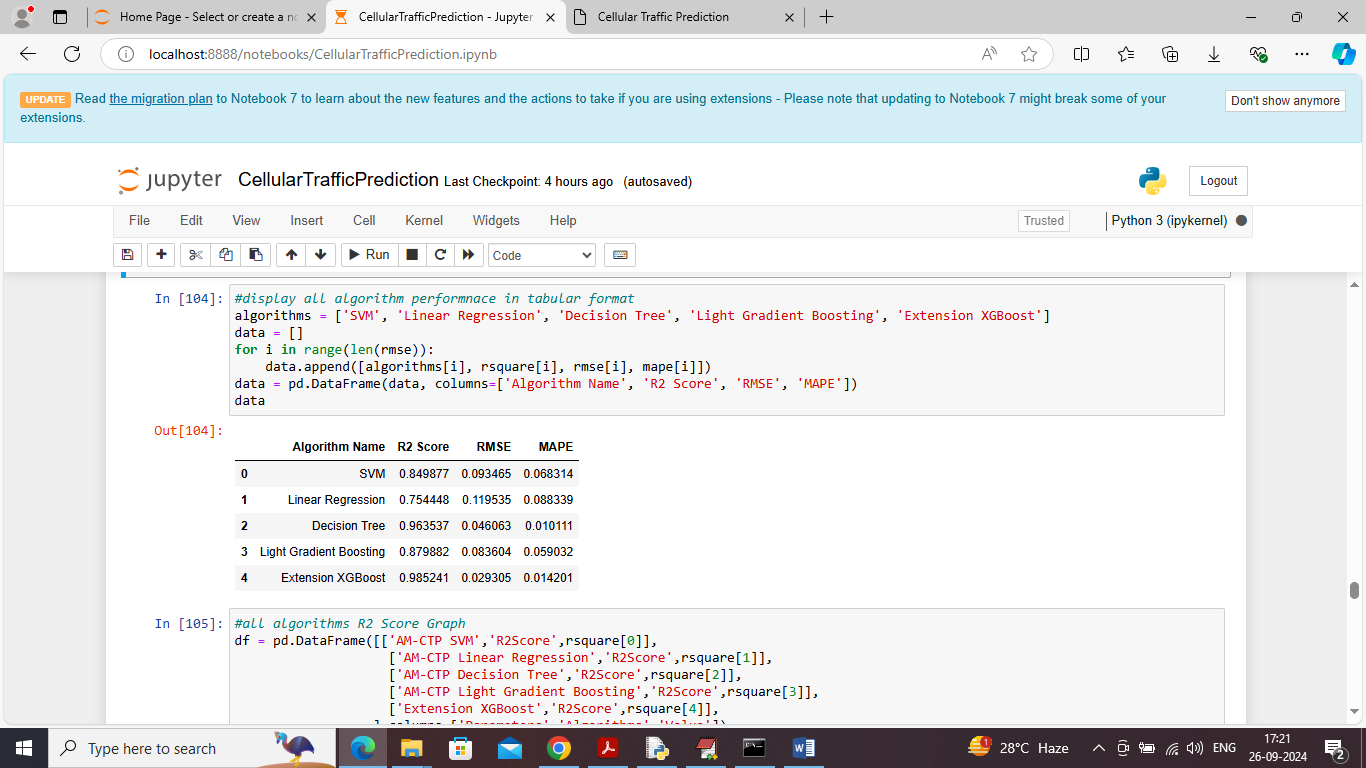
In above screen light gradient boosting got 87% r2 score and can see other metrics with prediction graph



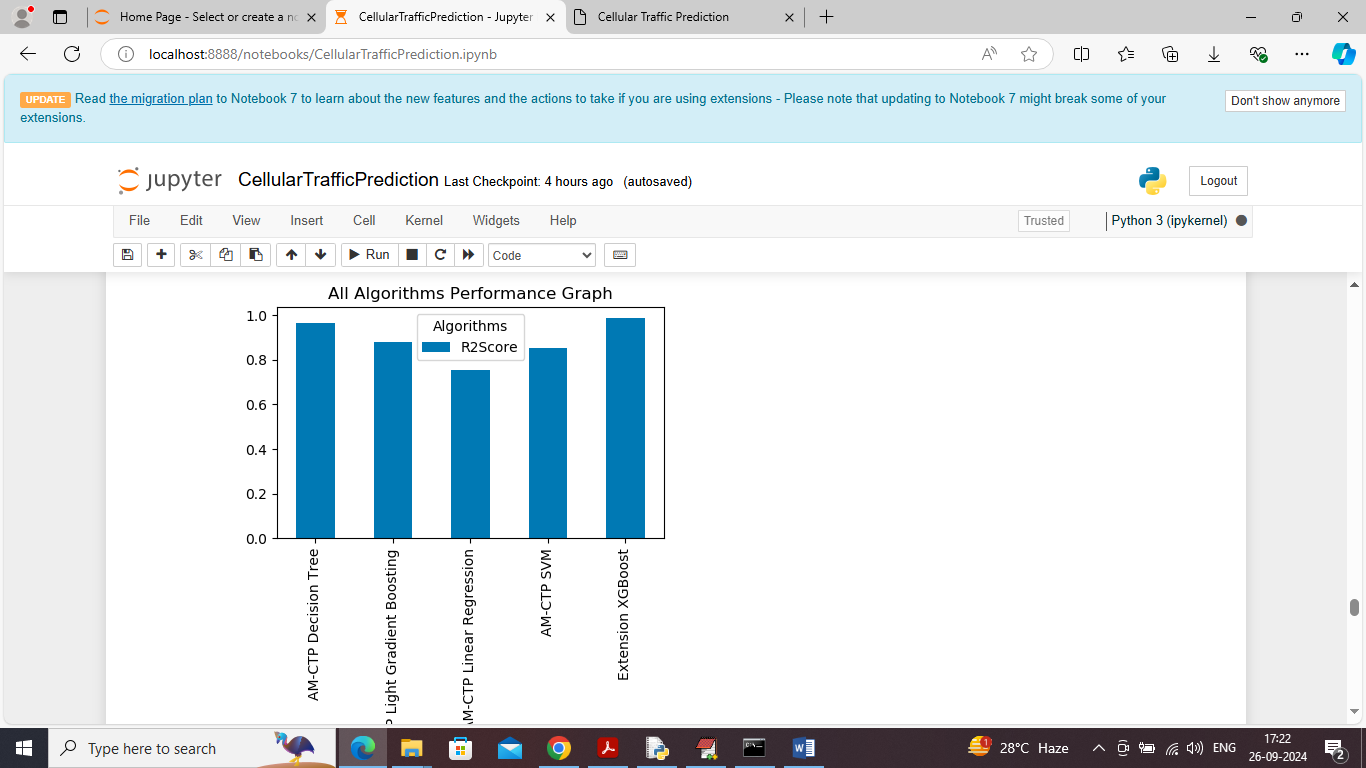
In above screen showing all propose models metrics in tabular format and can see among all algorithms Decision Tree got highest R2score and low RMSE and MAPE error so the selected model can be Decision Tree



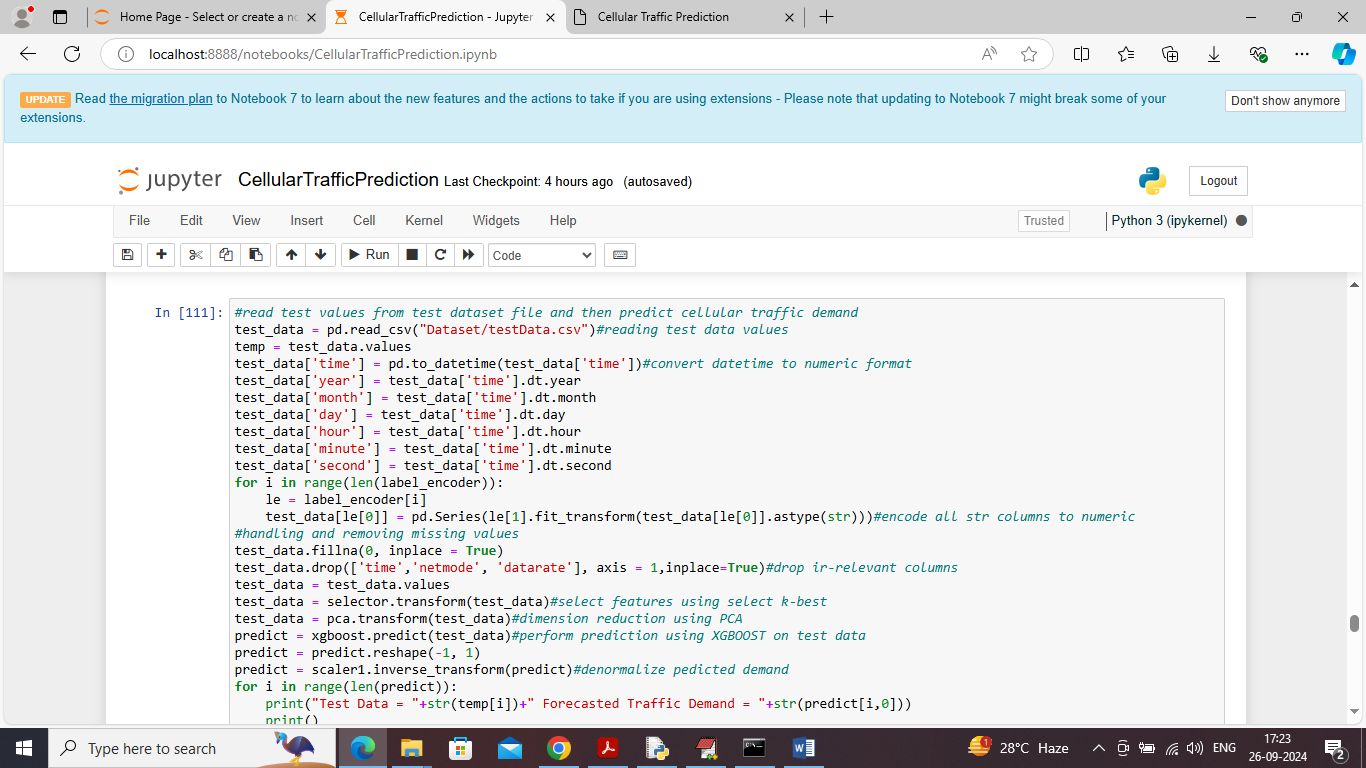
In above screen training extension XGBOOST algorithm with tuned estimators and it got 98% R2score which is higher than propose algorithms



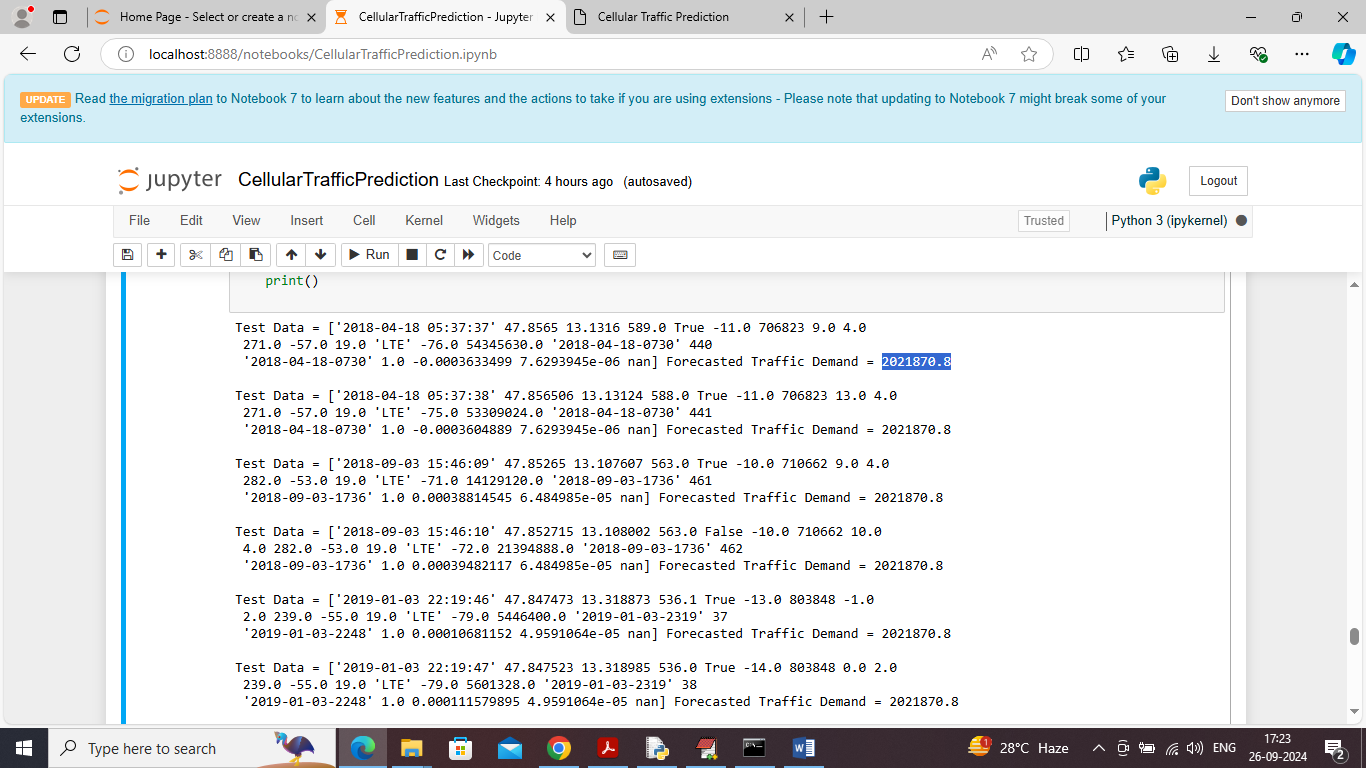
In above screen in table format can see all propose models algorithms performance along with extension algorithm



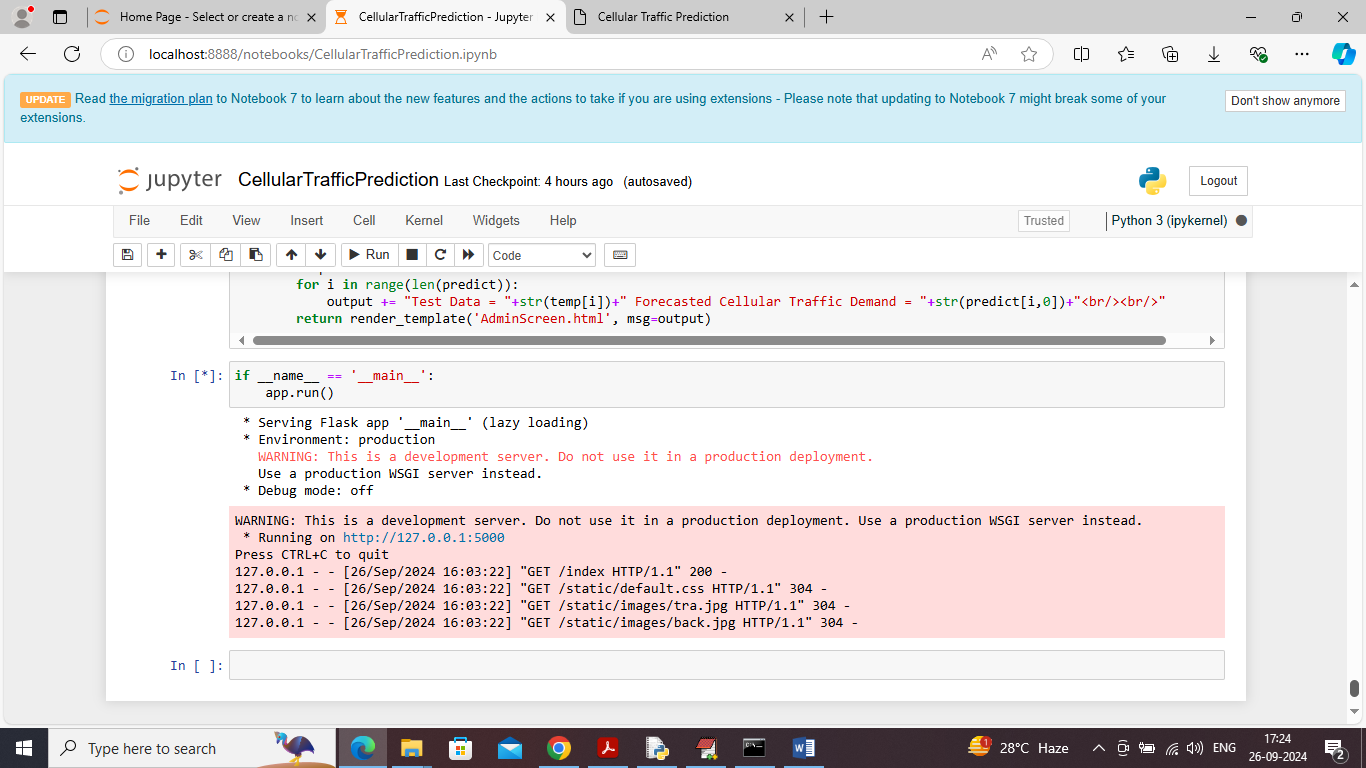
In above screen displaying all algorithms R2score comparison graph between all algorithms and in all algorithms Decision Tree and extension XGBOOST got high R2score



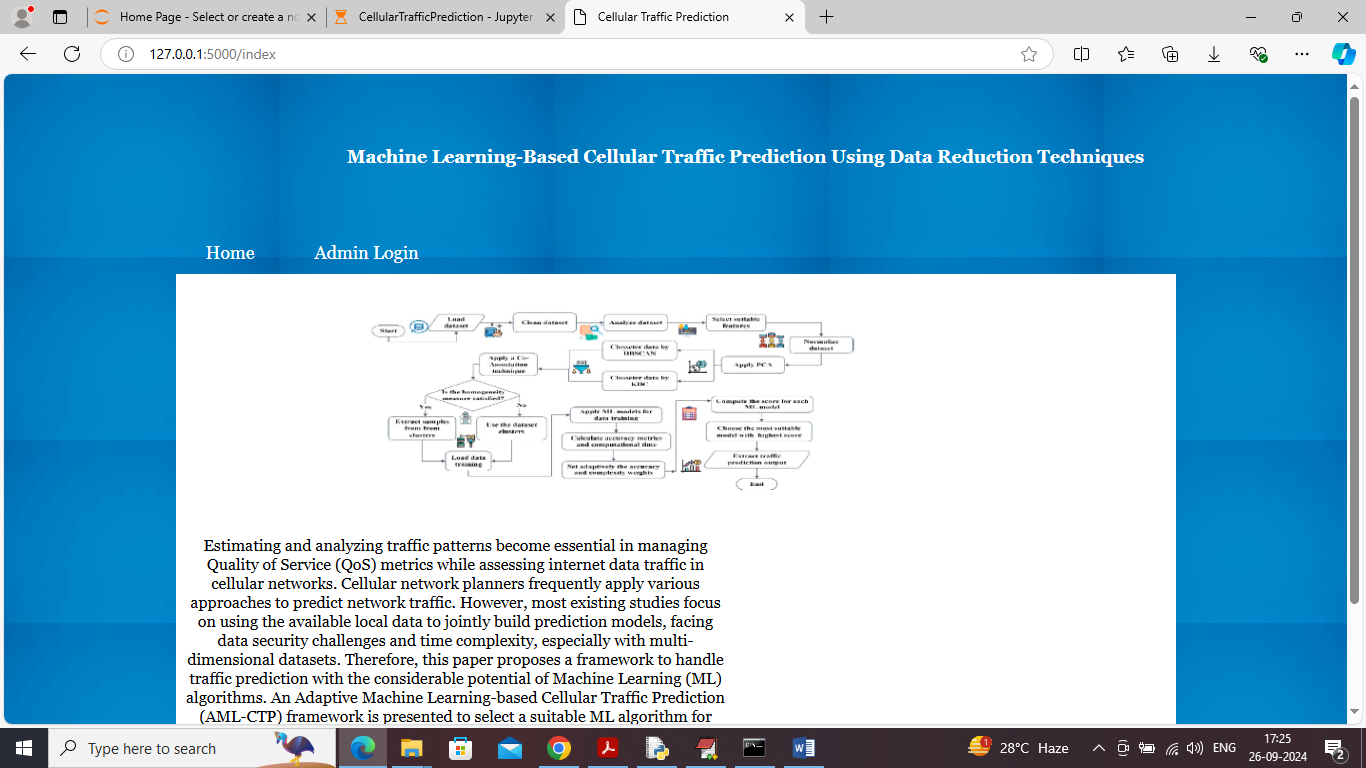
In above screen defining prediction code which will read test data file and then apply all processing techniques and then apply XGBOOST extension algorithm to forecast cellular traffic



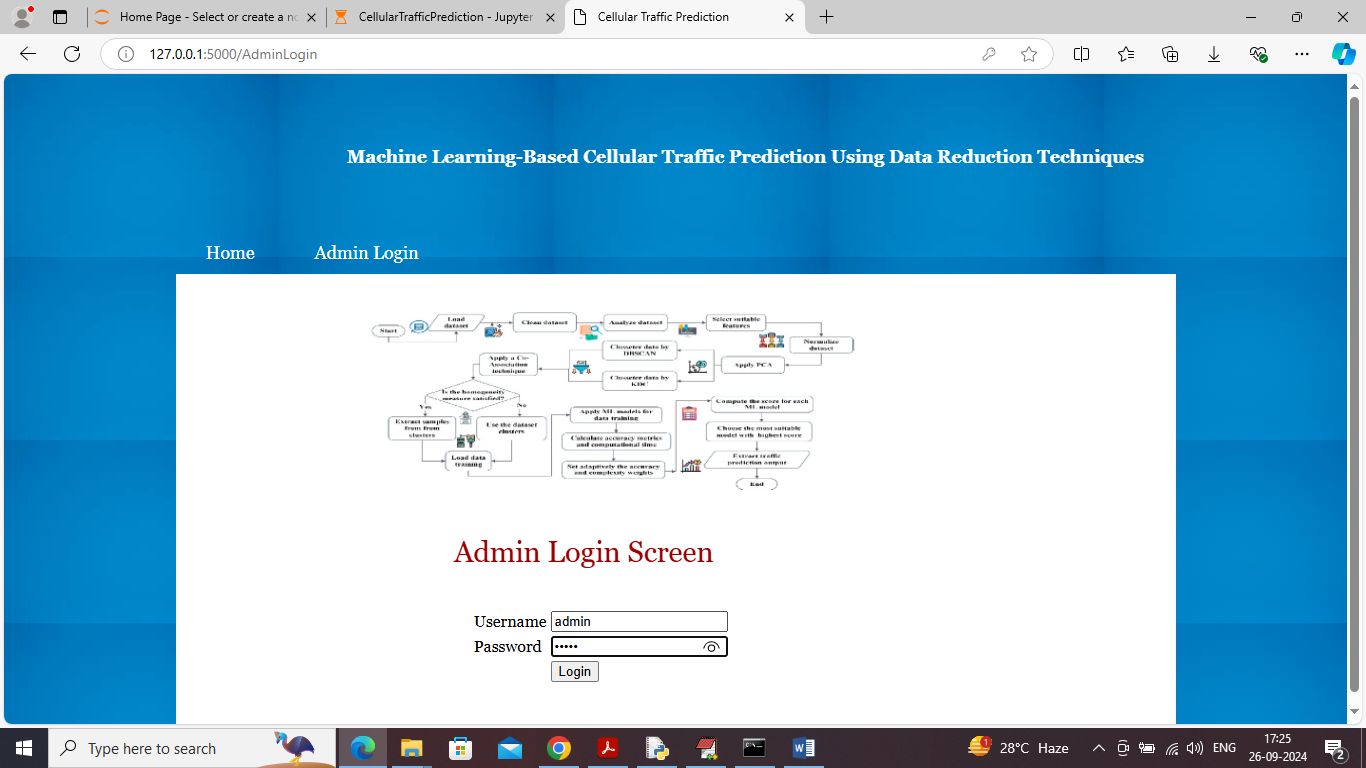
In above screen in square bracket we can see test data values and after square bracket we can see predicted Cellular traffic rate



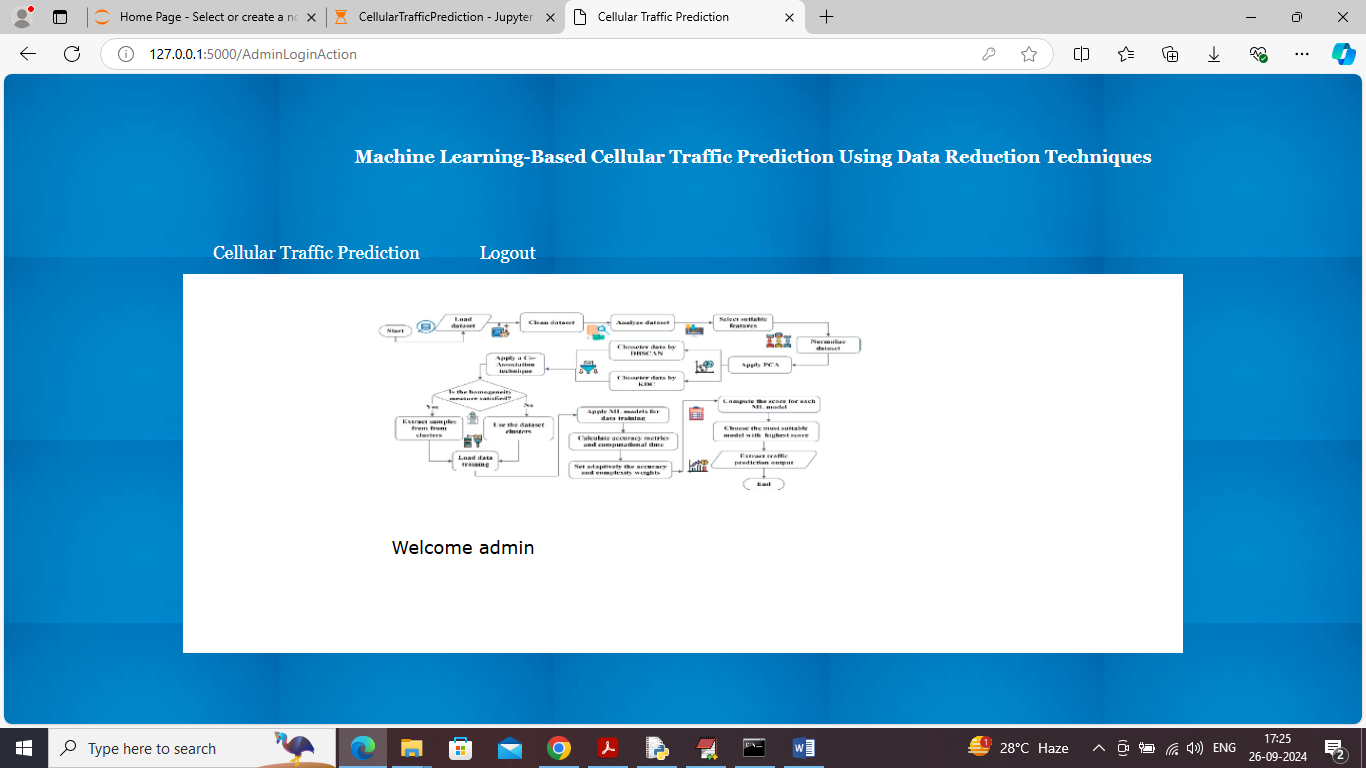
In above screen run all flask blocks code to start flask server and then open browser and enter URL as <http://127.0.0.1:5000/index> and press enter key to get below page



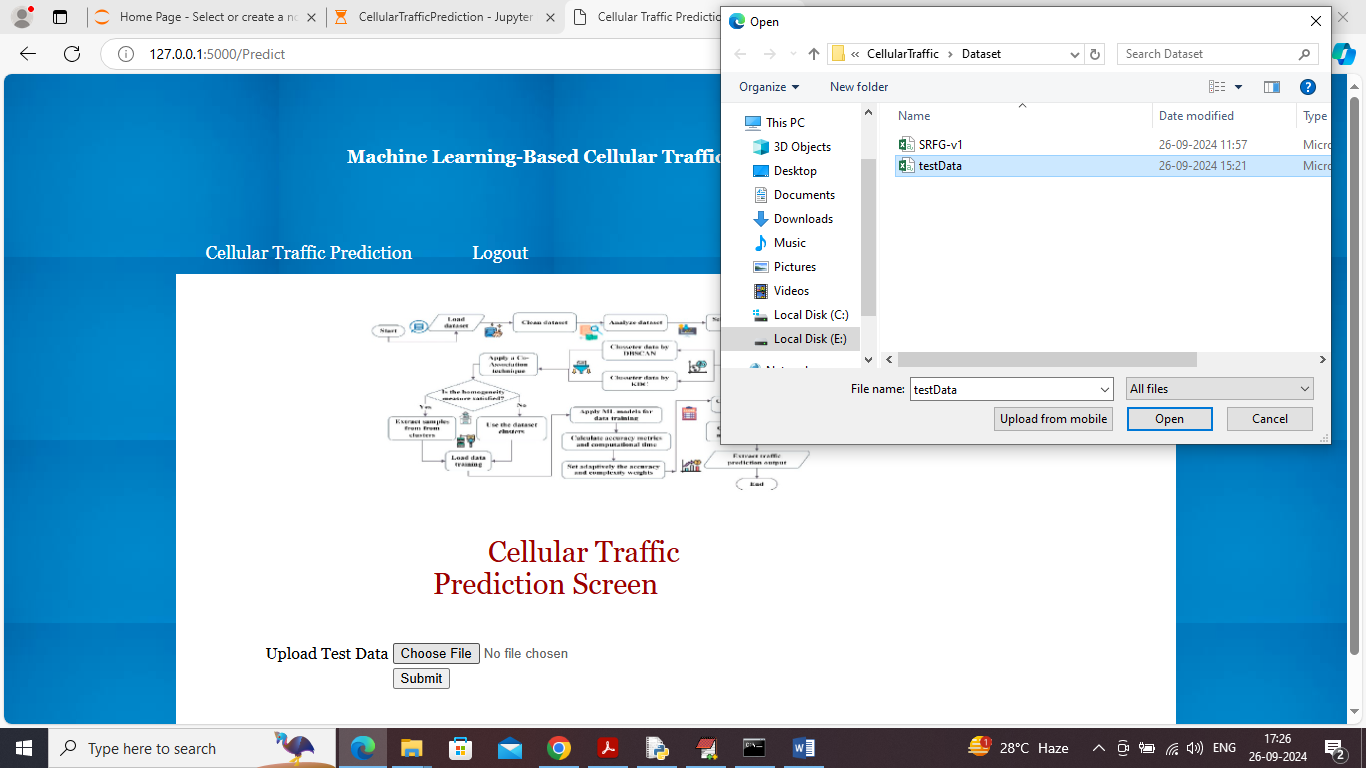
In above screen click on ‘Admin Login’ link to get below page



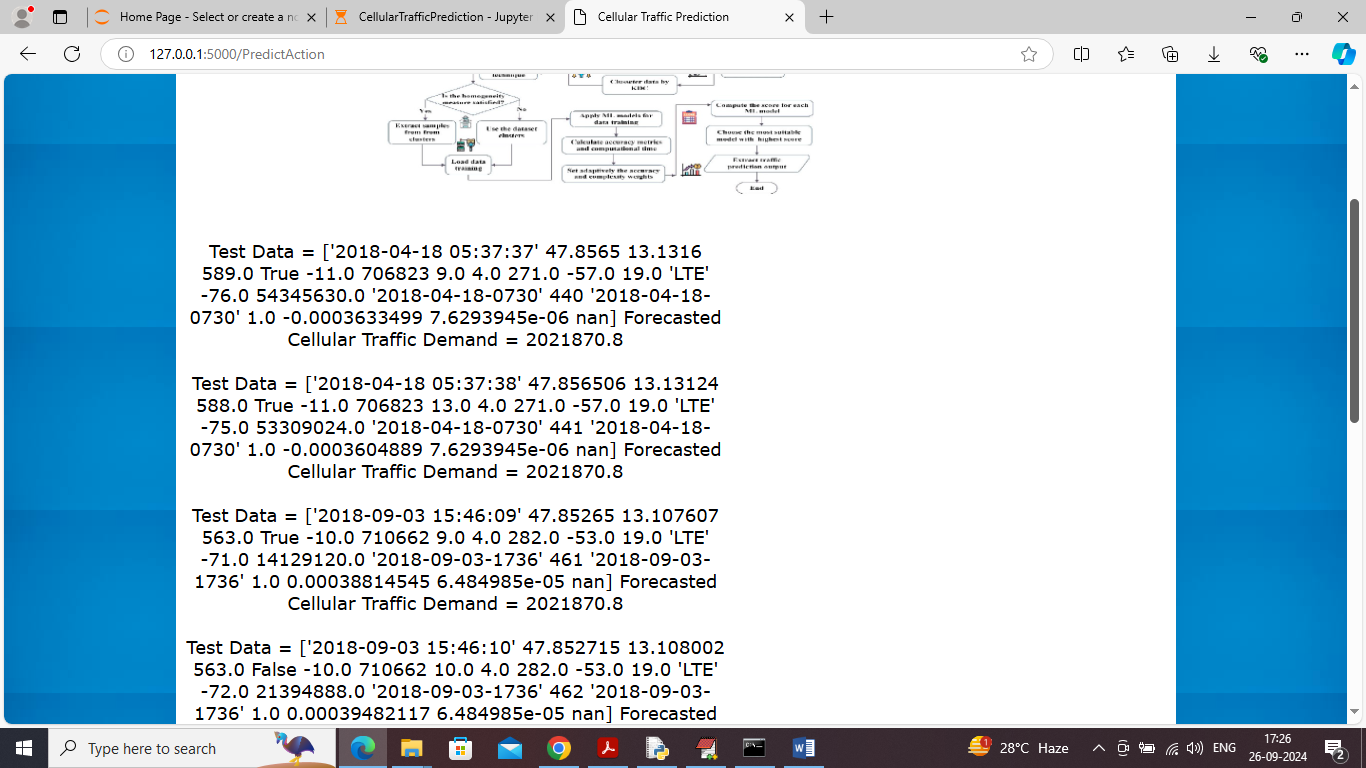
In above screen admin is login and after login will get below page



In above screen click on ‘Cellular Traffic Prediction’ link to get below screen



In above screen selecting and uploading test data file and then click on ‘Open’ button to get below page



In above screen can see test data values along with predicted cellular traffic rate