PUNE INSTITUTE OF COMPUTER TECHNOLOGY
DHANKAWADI, PUNE

DATA ANALYTIC MINI-PROJECT REPORT
ON

“USER CLASS PREDICTION FROM TRIP HISTORY ”

SUBMITTED BY

Maitraya Kakade	41427
Pranav Kulkarni	41430

Under the guidance of
Prof. Deepali Kadam

DEPARTMENT OF COMPUTER ENGINEERING
Academic Year 2020-21

Contents

1 Problem Statement

Use trip history dataset that is from a bike sharing service in the United States. The data is provided quarter-wise from 2010 (Q4) onwards. Each file has 7 columns. Predict the class of user.

2 Abstract

Data Analytic is the process of analyzing data and drawing conclusion from it.One such dataset is the trip history of capital bikeshares, which logs the travel history of its riders. The dataset is available for each quarter after 2010. We select the first quarter of 2017 for our analysis. The main goal is predict the class of the user as Member or Casual. We inspect various algorithms to achieve this goal and compare their performance.

3 Hardware and Software Requirements

3.1 Hardware Requirements

1. 500 GB HDD
2. 4GB RAM
3. Monitor
4. Keyboard

3.2 Software Requirements

1. 64 bit Open Source Operating System like Ubuntu 18.04
2. Python 3
3. Libararies like sklearn, pandas, matplotlib

4 INTRODUCTION

The data includes:

1. Duration – Duration of trip
2. Start Date – Includes start date and time
3. End Date – Includes end date and time
4. Start Station – Includes starting station name and number
5. End Station – Includes ending station name and number
6. Bike Number – Includes ID number of bike used for the trip
7. Member Type – Indicates whether user was a ”registered” member (Annual Member, 30-Day Member or Day Key Member) or a ”casual” rider (Single Trip, 24-Hour Pass, 3-Day Pass or 5-Day Pass)

This data has been processed to remove trips that are taken by staff as they service and inspect the system, trips that are taken to/from any of “test” stations at warehouses and any trips lasting less than 60 seconds (potentially false starts or users trying to re-dock a bike to ensure it’s secure).

We first preprocess the data. The Data fields are used to extract year, month, day, hour, week, minute and second. We add this as features to our dataset. Out of this we keep only month and hour field as rest fields have a uniform distribution across entire dataset.

We perform one hot encoding of the remaing data fields and use Random Forest Classifier with default parameters as our classificaton algorithm. We use train test split of 60-40 and report a classification accuracy of 90 %.

5 OBJECTIVE

- To analyse trip history dataset
- To predict the class of the user from given dataset..

6 Scope

We select only the first quarter of 2017 for our analysis. This is because the size of whole dataset makes it difficult to run the model on limited memory.

7 System Architecture

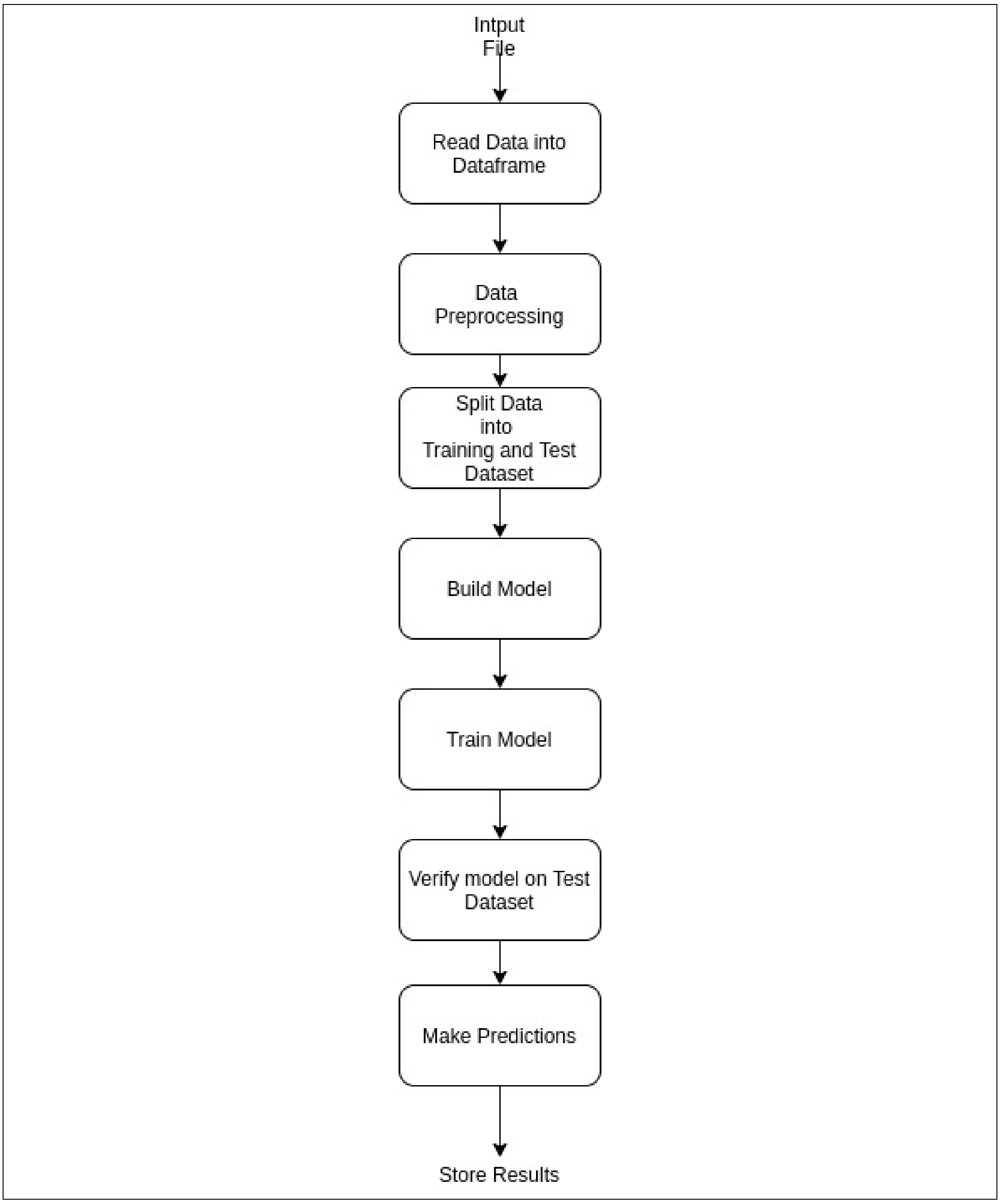
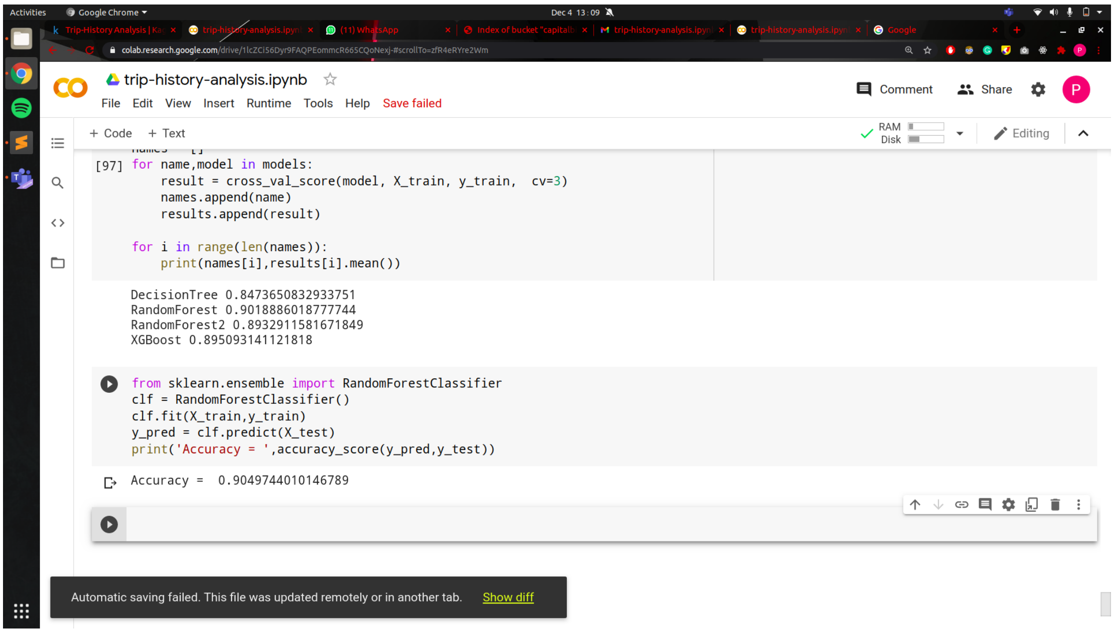


Figure 1: System Architecture

8 Test Cases


trip-history-analysis.ipynb
File Edit View Insert Runtime Tools Help Save failed
Comment Share
+ Code + Text
RAM Disk
Editing

```
[97] for name,model in models:
         result = cross_val_score(model, X_train, y_train,  cv=3)
         names.append(name)
         results.append(result)

     for i in range(len(names)):
         print(names[i],results[i].mean())
```

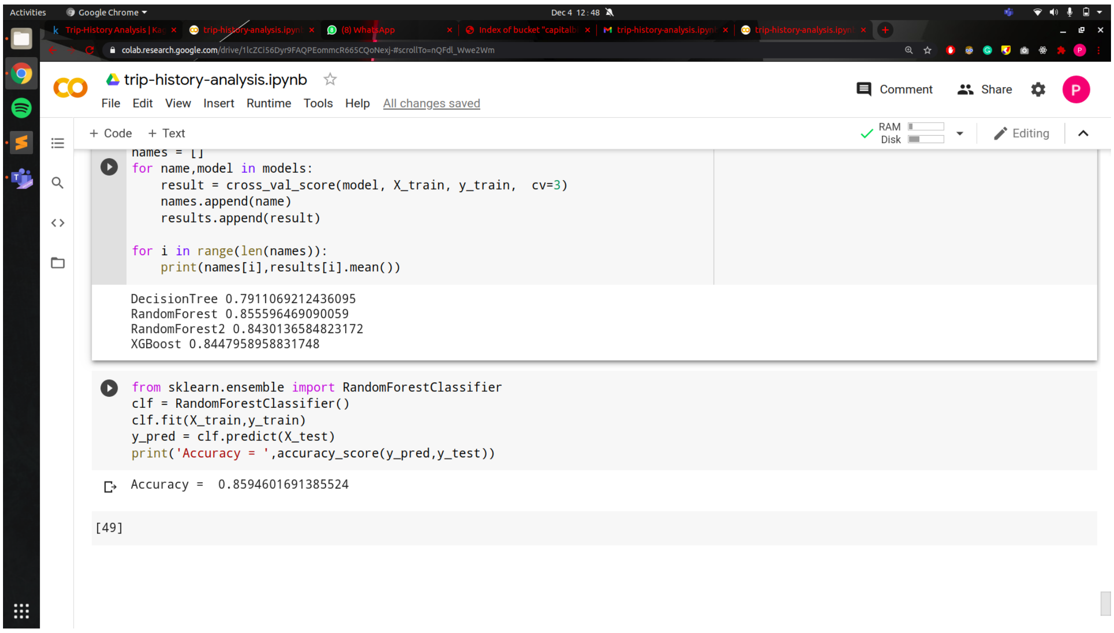
```
DecisionTree 0.8473650832933751
RandomForest 0.9018886018777744
RandomForest2 0.8932911581671849
XGBoost 0.895093141121818
```

```
from sklearn.ensemble import RandomForestClassifier
clf = RandomForestClassifier()
clf.fit(X_train,y_train)
y_pred = clf.predict(X_test)
print('Accuracy = ',accuracy_score(y_pred,y_test))
```

```
Accuracy =  0.9049744010146789
```

Automatic saving failed. This file was updated remotely or in another tab. Show diff

Figure 2: Output for the 2017 quarter 1 data


trip-history-analysis.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment Share
+ Code + Text
RAM Disk
Editing

```
names = []
for name,model in models:
    result = cross_val_score(model, X_train, y_train,  cv=3)
    names.append(name)
    results.append(result)

for i in range(len(names)):
    print(names[i],results[i].mean())
```

```
DecisionTree 0.7911069212436095
RandomForest 0.855596469090059
RandomForest2 0.8430136584823172
XGBoost 0.8447958958831748
```

```
from sklearn.ensemble import RandomForestClassifier
clf = RandomForestClassifier()
clf.fit(X_train,y_train)
y_pred = clf.predict(X_test)
print('Accuracy = ',accuracy_score(y_pred,y_test))
```

```
Accuracy =  0.8594601691385524
```

```
[49]
```

Figure 3: Output for the 2017 quarter 2 data

9 Result

The Cross Validation Scores for Various models are:

Model	Cross-Validation Score
DecisionTree	0.8464524910674235
RandomForest	0.9051857924342496
RandomForest2	0.8935541084695777
XGBoost	0.896276417482586

Table 1: Cross Validation Scores for vaious Models

We see that Random Forest Classifier gives the best score. We then use this model to perform training and testing of the model. After training, the model gives an accuracy of 90.86 %.

10 Conclusion

We presented classification of trip hsitory dataset to predict the clas of user using Random Forest Classifier. We report an classification accuracy of 90%.

References

[1] https://www.capitalbikeshare.com/system-data

[2] https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html