Social Media Data using Various Classification Algorithms in Datamaning

Deepa B, JeenMarseline K.S

Abstract— Data Mining is one of the most successful domains in research. It describes the past and speculates the future for analysis. There are several techniques used in data mining. Among them classification is one of the main data mining techniques based on machine learning. In classification technique data set is classified into predefined set of groups or classes. Mathematical techniques such as decision tree, linear regression, neural networks and statistics are used for classification methods. Classification is a problem to identify which set of categories the new observation belongs to using training data set. This paper analyses the data taken from social media and uses the classification algorithm for making a comparative study on social advertisement using python.

Keywords — Data Mining, Classification Algorithm, Linear Regression, Decision Tree, Python.

I. INTRODUCTION

Data Mining is a task to bring out required data from large database. It is the process of finding the hidden details from repositories. In that pattern finding and data analysis is a regulation that deals with the help of machine learning. The model to be built depending on the type of data used and learning problem is divided into supervised and unsupervised learning. In supervised learning the training set consist of input x and output y but unsupervised learning consists of only input x without an output .

In that predictive data mining classification is one of the techniques. In that classification using social networks advertisement data set is used to predict the data based on some attributes.

II. METHODOLOGY

The proposed work done with the help of python Language and Scikit learn datamining package used for build the model using logistic regression and decision tree classifier algorithm.

III. DATA SET DESCRIPTION

Data set provides the customer information on social media consists of 5 attributes. The attributes are User Id, Gender, Age, Estimated Salary, Purchased. In that dataset Purchased is a dependent variable and remining are independent variable. It deals about the person who are all view the advertisement and buy the product, according to that to predict the new customer belongs to which class (i.e. Dependent variable) using the feature values (i.e. Independent variable).

IV. LIBRARIES USED

The model is built using some python standard libraries taken from python community. The libraries are NumPy, pandas, matplotlib, seaborn, and sklearn. Among these pandas used for data analysis with series and Data frame. NumPy is called as numerical python used for scientific calculation. Matplotlib and seaborn is used for data visualization in the form of plots. Model built done with the help of sklearn package.

V. TRAINING OF CLASSIFIER

During model building the dataset is split as training set and Test set. To find the relationship between features and classes the classifier should be trained using training dataset then produce the output in the form of classification report, Accuracy score and confusion matrix.

VI. TESTING OF CLASSIFIER

After train the model using classification algorithm the design used to test the new data for predicting Future classes and calculate the Accuracy based on the test data.

VII. ALGORITHMS USED

A. *Logistic Regression*

Classification is a statistic and machine learning techniques used to identify the problem in which the new data belongs to which set of categories is based on training data. Among these technique Logistic regression is one of the supervised learning algorithms to measure the relationship between the categorical dependent variable and one or more independent variable by using the logistic function

B. *Decision Tree Classifier*

Decision tree classifier is on the classification algorithm. Based on the criteria it divides the data set into some smaller subset. Several sorting criteria is used in decision tree classifier.

Steps:

1. Root node is created first
2. Calculate the entropy with current state H(S)
3. Calculate the entropy for each attribute which respect to the attribute x H (S, x)

Revised Manuscript Received on September 14, 2019.

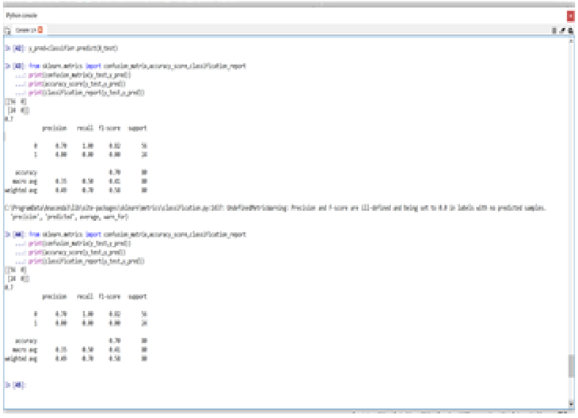
Deepa B, Assistant Professor Sri Krishna Arts and Science College Coimbatore, Tamilnadu, India.(Email: deepab@skasc.ac.in)

Dr. JeenMarseline K.S, Assistant Professor Sri Krishna Arts and Science College Coimbatore, Tamilnadu, India.(Email: jeenmarselineks@skasc.ac.in).

4. Maximum value Attribute is selected with respect to IG (S, x)
5. Remove the attributes from the set of attributes which offers highest IG
6. Repeat the process until Decision tree has all leaf nodes

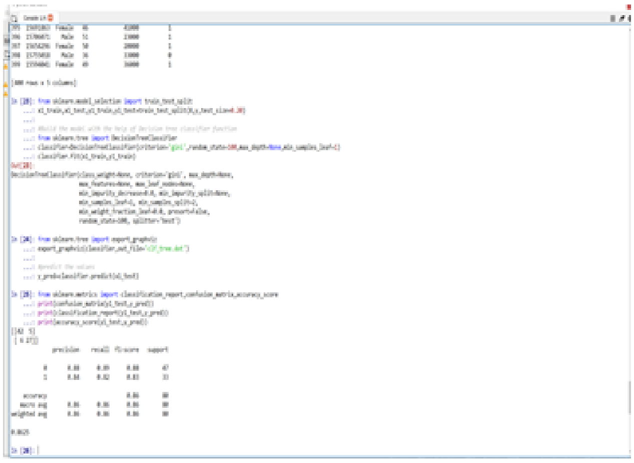
VIII. RESULTS AND DISCUSSION

To summarize the classification process, use confusion matrix it is in the form of matrix deals with classified verses misclassified. Classification report is produced in the form of precision, recall, F support. The following figures shows the classification report, confusion matrix and accuracy of Logistic regression and Decision tree classifier algorithms using python



```
In [43]: y_pred=classifier.predict(X_test)

In [43]: from sklearn.metrics import confusion_matrix,accuracy_score,classification_report
    ...: print(confusion_matrix(y_test,y_pred))
    ...: print(accuracy_score(y_test,y_pred))
    ...: print(classification_report(y_test,y_pred))
[[56  0]
 [24  0]]
0.7
              precision    recall  f1-score   support

           0       0.70      1.00      0.82        56
           1       0.00      0.00      0.00        24

    accuracy                           0.70        80
   macro avg       0.35      0.50      0.41        80
weighted avg       0.49      0.70      0.58        80

C:\ProgramData\Anaconda3\lib\site-packages\sklearn\metrics\classification.py:1437: UndefinedMetricWarning: Precision and F-score are ill-defined and being set to 0.0 in labels with no predicted samples.
  'precision', 'predicted', average, warn_for)

In [44]: from sklearn.metrics import confusion_matrix,accuracy_score,classification_report
    ...: print(confusion_matrix(y_test,y_pred))
    ...: print(accuracy_score(y_test,y_pred))
    ...: print(classification_report(y_test,y_pred))
[[56  0]
 [24  0]]
0.7
              precision    recall  f1-score   support

           0       0.70      1.00      0.82        56
           1       0.00      0.00      0.00        24

    accuracy                           0.70        80
   macro avg       0.35      0.50      0.41        80
weighted avg       0.49      0.70      0.58        80


In [45]:
```

Fig 1. Classification report of logistic regression



```
395  15691863  Female  46      41000     1
396  15706071    Male  51      23000     1
397  15654296  Female  50      20000     1
398  15755018    Male  36      33000     0
399  15594041  Female  49      36000     1

[400 rows x 5 columns]

In [28]: from sklearn.model_selection import train_test_split
    ...: x1_train,x1_test,y1_train,y1_test=train_test_split(X,y,test_size=0.20)
    ...:
    ...: #Build the model with the help of Decision tree classifier function
    ...: from sklearn.tree import DecisionTreeClassifier
    ...: classifier=DecisionTreeClassifier(criterion='gini',random_state=100,max_depth=None,min_samples_leaf=1)
    ...: classifier.fit(x1_train,y1_train)
Out[28]:
DecisionTreeClassifier(class_weight=None, criterion='gini', max_depth=None,
                       max_features=None, max_leaf_nodes=None,
                       min_impurity_decrease=0.0, min_impurity_split=None,
                       min_samples_leaf=1, min_samples_split=2,
                       min_weight_fraction_leaf=0.0, presort=False,
                       random_state=100, splitter='best')

In [29]: from sklearn.tree import export_graphviz
    ...: export_graphviz(classifier,out_file='c2f_tree.dot')
    ...:
    ...: #predict the values
    ...: y_pred=classifier.predict(x1_test)

In [30]: from sklearn.metrics import classification_report,confusion_matrix,accuracy_score
    ...: print(confusion_matrix(y1_test,y_pred))
    ...: print(classification_report(y1_test,y_pred))
    ...: print(accuracy_score(y1_test,y_pred))
[[42  5]
 [ 6 27]]
              precision    recall  f1-score   support

           0       0.88      0.89      0.88        47
           1       0.84      0.82      0.83        33

    accuracy                           0.86        80
   macro avg       0.86      0.86      0.86        80
weighted avg       0.86      0.86      0.86        80

0.8625

In [30]:
```

Fig 2 Clssification Report Of Decision Tree

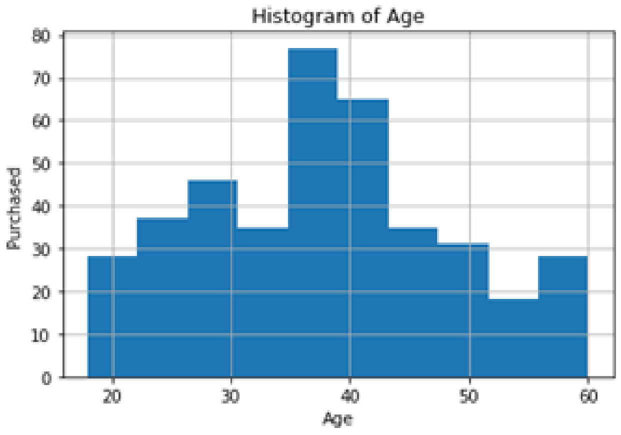


Fig 3 Histogram Using Logistic Regression

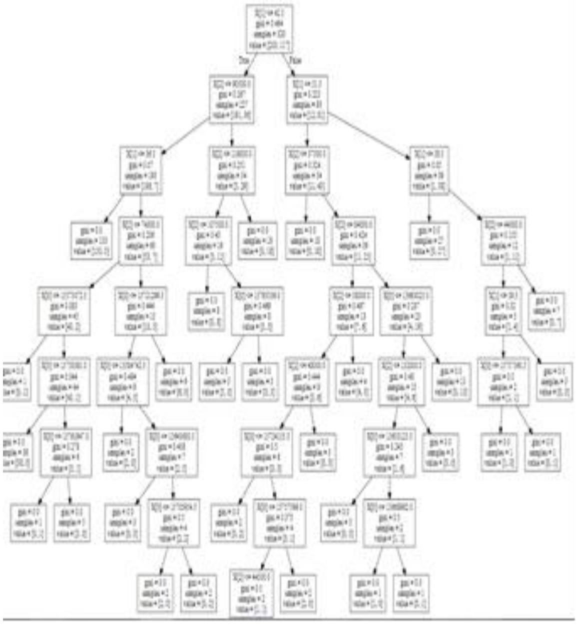


Fig 4 Decision Tree Based On Dataset

IX. CONCLUSION

The classification is done with the help of Logistic regression and decision tree classifier. In that decision tree produce the accuracy 86% than the Logistic regression. So that Decision tree is a best model for predicting data than Logistic regression.

REFERENCES

1. Pratiyush Guleria, Manu Sood Department of Computer Science Himachal Pradesh University Shimla, INDIA pratiyushguleria@gmail.com soodm_67@yahoo.com Predictive Data Modeling: Educational Data Classification and Comparative Analysis of Classifiers Using Python
2. Mr. Santhana Krishnan PG Student Department of Computer Applications Anna University, BIT Campus Tiruchirapalli-24 Santhanakrish1996 @gmail.com Dr. Geetha. S Asst. Professor Department of Computer Applications Anna University, BIT Campus Tiruchirapalli-24 Kasagee1971@gmail.com Prediction of Heart Disease Using Machine Learning Algorithms
3. Fabien Dubosson*, Stefano Bromuri,*† and Michael Schumacher* *AISLab, HES-SO Valais//Wallis †Management, Science and Technology, Open University of the Netherlands {fabien. dubosson, michael. schumacher} @hevs.ch stefano.bromuri @ou.nl A Python Framework for Exhaustive Machine Learning Algorithms and Features Evaluations
4. Jaswitha Abbineni1 Information technology Vijayawada, Vrsec India abbinenisaipriya@gmail.com Ooha Thalluri 2 Information technology Vrsec Vijayawada, India oohatalluri96@gmail.com Software Defect Detection using Machine Learning Techniques
5. Bhawana Tyagi School of Information Technology C-DAC Noida, India bhawana1988@gmail.com Rahul Mishra School of Information Technology C-DAC Noida, India rahulmishra@cdac.in Neha Bajpai School of Information Technology C-DAC Noida, India nehakapoor@cdac.in Machine Learning Techniques to Predict Autism Spectrum Disorder.