USER RESPONSE PREDICTION SYSTEM USING MACHINE LEARNING TECHNIQUES

Nabanita Chatterjee
The Northcap University

INTRODUCTION

- The online advertising industry is now a multi-billion-dollar sector, with accurate prediction of ad click-through rates (CTR) being crucial for advertisers and search engines.
- Click-through rate (CTR) measures the percentage of impressions that result in a click on an ad.
- Search ads are displayed when a user searches for a specific keyword.
- Paid search advertising, also known as Pay-per-click (PPC) advertising, involves advertisers paying a bid amount to have their ads displayed when users search for particular keywords.
- Predicting CTR helps advertisers and search engines determine which ads to display and optimize their ad placements for better performance.
- Machine learning is a common approach used to predict ad click-through rates accurately.
- Accurate CTR prediction enables advertisers to optimize their ad campaigns, allocate budgets more efficiently, and improve return on investment (ROI).
- Search engines use CTR prediction to display relevant and engaging ads to users, enhancing the overall user experience and driving more ad revenue.
- CTR prediction models take into account various factors such as ad content, keyword relevance, user behavior, and historical performance data to make accurate predictions.
- By predicting CTR, advertisers can adjust their bids and ad creatives in real-time to ensure their ads reach the right audience and achieve their advertising goals.

PROBLEM STATEMENT

In this project, I have worked with advertising data from a marketing agency to build a machine-learning algorithm that predicts whether a user will click on an advertisement.The dataset contains 10 variables:

- 'Daily Time Spent on Site
- 'Age'
- 'Area Income'
- 'Daily Internet Usage'
- 'Ad Topic Line'
- 'City'
- 'Male'
- 'Country'
- 'Timestamp'
- 'Clicked on Ad'

PROBLEM STATEMENT

- The main focus is on the 'Clicked on Ad' variable, which can take two values: 0 and 1. A value of 0 indicates that a user did not click on the advertisement, while a value of 1 means that the user clicked on the ad.
- I aim to use the other 9 variables to accurately predict the 'Clicked on Ad' variable. By doing so, I can determine which factors influence the user's decision to click on an ad.
- Additionally, I will conduct exploratory data analysis to investigate how the combination of 'Daily Time Spent on Site' and 'Ad Topic Line' impacts the likelihood of a user clicking on the ad. This analysis will help us gain insights into user behavior and ad effectiveness in relation to these variables.

OBJECTIVES

The primary objectives of this project are to conduct a comprehensive exploration of the advertising data, perform quantitative analysis, and utilize machine learning techniques to make predictions based on the data. The table below provides a detailed description of the features present in the dataset:

1.Daily Time Spent on a Site: This feature represents the amount of time (in minutes) that a user spends on the website.

2.Age: The age of the customer expressed in terms of years.

3.Area Income: This feature indicates the average income level of the geographical area where the consumer is located.

4.Daily Internet Usage: The average number of minutes per day that the consumer spends on the internet.

5.Ad Topic Line: The headline or title of the advertisement.

6.City: The city where the consumer is located.

7.Male: A binary variable indicating whether or not the consumer is male (1 for male, 0 for female).

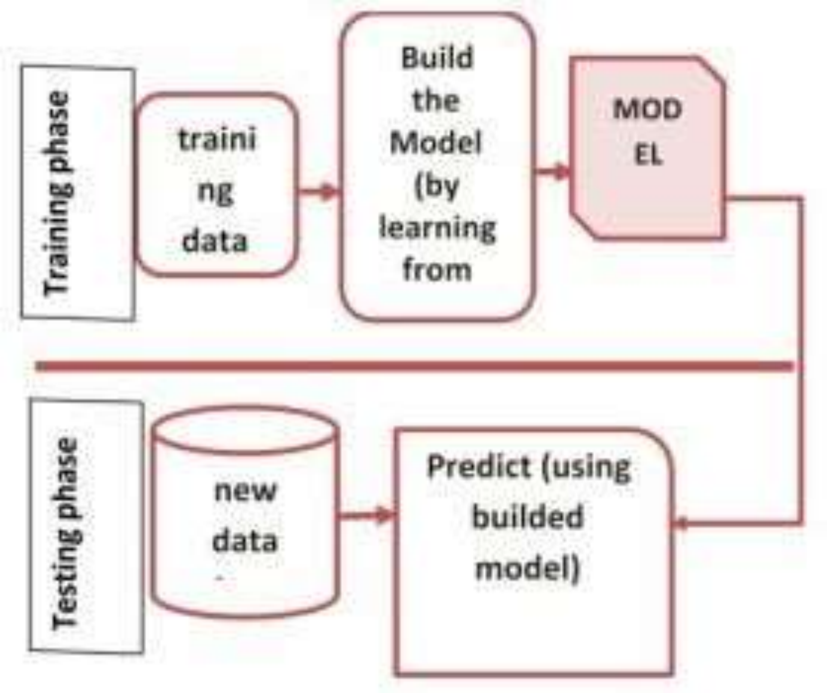
8.Country: The country where the consumer is located.

9.Timestamp: The time at which the user either clicked on an ad or closed the ad window.

10.Clicked on Ad: This is the target variable we aim to predict, with values 0 or 1 indicating whether the user clicked on the advertisement or not.

Throughout the project, we will delve into the data to gain a deeper understanding of advertising trends, patterns, and consumer behavior. We will conduct quantitative analyses to derive meaningful insights from the data. Additionally, we will apply various machine learning techniques to develop predictive models for determining the likelihood of a user clicking on an ad based on the given features. By accomplishing these objectives, we aim to enhance our understanding of advertising effectiveness and optimize ad targeting strategies.

ARCHITECTURE


Training phase
traini ng data
Build the Model (by learning from
MOD EL
Testing phase
new data
Predict (using builded model)

OVERVIEW

The entire process is divided into 7 major steps :

1. Importing dependencies and loading Data set
2. Data Preprocessing
3. Exploratory Analysis
4. statistical Analysis
5. Train Test Split
6. Training the Model
7. Testing the model accuracy

The data set was provided from Kaggle (attached)

STEP 1:IMPORTING DEPENDENCIES AND LOADING DATASET

In order to do the predictive analysis we need to import some python libraries which will help in data visualization, dealing with data set and will also provide pre-implemented Machine Learning models.

STEP 2 :DATA PREPROCESSING

- In this project, data cleaning was performed manually by identifying relationships between multiple columns to ensure accuracy.
- Categorical columns, such as "Ad Topic Line," "City," and "Country," were identified and analyzed for unique values.
- The "Ad Topic Line" column contained all unique values, making it unsuitable for prediction due to the lack of data patterns.
- The "City" column had 969 unique values out of 1000, making it challenging to work with for prediction purposes, and it was also omitted from further analysis.
- The "Country" column had one repeated unique element (France) occurring nine times.
- Countries with the highest number of visitors were determined and listed in the DataFrame.
- The "Timestamp" category, representing the exact time of ad clicks, was analyzed and expanded into four new categories: month, day of the month, day of the week, and hour.
- Creating these new variables allows the ML model to process and discover possible dependencies and correlations in the data.
- The original "Timestamp" variable was removed from the table after creating the new date-related variables.
- The "Day of the week" variable contains values from 0 to 6, representing each day of the week (from Monday to Sunday).

Step 3: Exploratory Data Analysis

We have performed the following analysis.

1. Distribution of daily time with ads
2. Distribution of daily internet with ads
3. Top cities with daily time
4. Top Cities with area income
5. Top Cities with avg Internet
6. Investigating the Country Variable
7. Top city with avg internet
8. Top countries with daily time
9. Top Cities with area income

STEP 4 : STATISTICALANALYSIS

We have performed following analysis.

1. Examine the data
2. Data type and length of the variables
3. Check for Missing Values
4. Numerical and Categorical Variables Identification
5. Summarizing Numerical Variables
6. Summarizing Categorical Variables
7. Categorizing Quantitative and Qualitative Variables
8. Outliers
9. Identifying Potential Outliers using IQR
10. T-Test & F-Test Between Groups of People that Clicked on Ads
11. Variance
12. Mean
13. Testing for Normality
14. Mann-Whitney U Test

STEP 5 :TRAIN AND TEST DATA SETS

- Once the dataset is processed,we need to divide it into two parts:training and test set.We will import and use the train_test_split function for that.All variables except 'Clicked on Ad' will be the input values X for the ML models.The variable 'Clicked on Ad' will be stored in y,and will represent the prediction variable.

- X_train and Y_train are used to train the Machine Learning model while x_test is used as input for making predictions which will be then validated with the y_test values.

STEP 6 :TRAINING THE MACHINE LEARNING MODEL

Indeed, based on the output data, we can observe that there are two categories:

1. The customer will click on the ad (represented by 1)

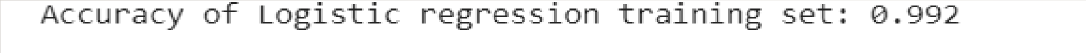
2. The customer won't click on the ad (represented by 0)

3. This clearly indicates that we are dealing with a classification problem in this project. The task is to classify instances into one of the two classes based on the input features.

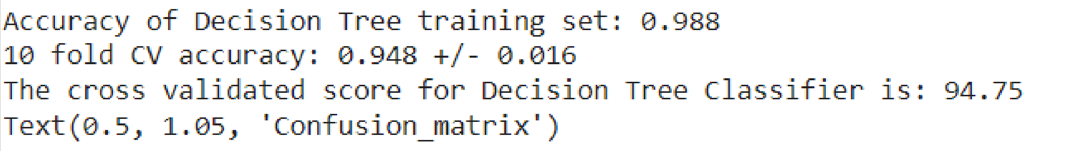
Furthermore, during the visualization of the data, we observed decision boundaries, suggesting that the data may have separable regions that can be used as a basis for selecting an appropriate Machine Learning model for classification. These decision boundaries can help us determine how to best separate the data points belonging to different classes, making it suitable for classification algorithms to learn and make accurate predictions. With this insight, we can proceed to explore and implement various classification models to predict whether a customer will click on the ad or not based on the given features.

STEP 7 :CHECKING MODELACCURACY

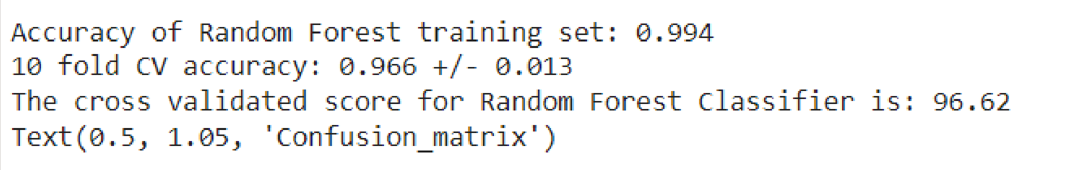
Final step is to check the accuracy of the Machine Learning model which we have created for ad click prediction :



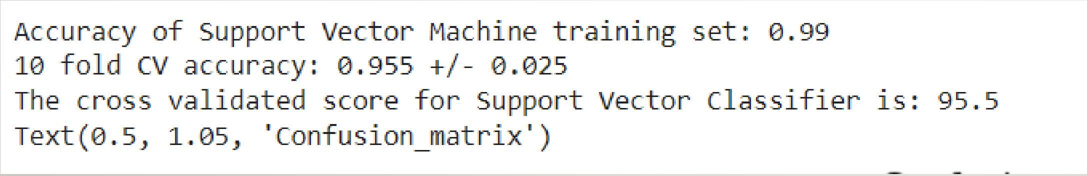
```
 Accuracy of Logistic regression training set: 0.992
```



```
Accuracy of Decision Tree training set: 0.988
10 fold CV accuracy: 0.948 +/- 0.016
The cross validated score for Decision Tree Classifier is: 94.75
Text(0.5, 1.05, 'Confusion_matrix')
```



```
Accuracy of Random Forest training set: 0.994
10 fold CV accuracy: 0.966 +/- 0.013
The cross validated score for Random Forest Classifier is: 96.62
Text(0.5, 1.05, 'Confusion_matrix')
```



```
Accuracy of Support Vector Machine training set: 0.99
10 fold CV accuracy: 0.955 +/- 0.025
The cross validated score for Support Vector Classifier is: 95.5
Text(0.5, 1.05, 'Confusion_matrix')
```

CONCLUSION

During the experimentation phase, the Random Forest model showed promising results with good precision. Its model fitting time was relatively quick, allowing for efficient tuning of parameters. On the other hand, the Linear Kernel Support Vector Classifier (SVC) took significantly longer to provide results. However, it had shorter prediction times compared to both Random Forest and k Nearest Neighbors classifiers.

While the SVC required only one parameter to be tuned, its AUC (Area Under the Curve) performance was slightly higher than the Random Forest's. In contrast, the k Nearest Neighbors model performed the worst in terms of AUC and prediction time, making it less suitable for this specific dataset.

The final choice for the model is the Linear SVC due to its slightly higher AUC and faster prediction times compared to the Random Forest model. The overall accuracy achieved at the end of this project is 96%, which is still quite high but not as good as the Random Forest or the previous Support Vector Classifier's results. The decision to choose Linear SVC may have been influenced by the trade-off between accuracy and prediction time, as well as the specific requirements of the application.