



# Google ads Hourly Analysis

Date : 20-06-2023

Project Start Date - End Date	<ul style="list-style-type: none"><li>● Start Date – 20 -06 -2023</li><li>● End Date – 20-06 2023</li></ul>
Objectives	<ul style="list-style-type: none"><li>● To analyses how many people who clicked on the advertisement enrolled in our course</li><li>● General exploratory analyses</li><li>● General descriptive analyses</li></ul>
Milestones accomplished the week of Start Date - End Date:	<ul style="list-style-type: none"><li>● Descriptive analyses</li><li>● Exploratory analyses</li><li>● Classification of data with respect to term</li></ul>

## Contact Information

---

This project is performed for educational purpose of under the guidance of Siddhivinayak Sir .

### Project Manager

Name : Siddhivinayak Phulwadkar

Mobile: 9028965955

Email:

[siddhivinayakphulwadkar@gmail.com](mailto:siddhivinayakphulwadkar@gmail.com)

### Student Name

Name : Mahesh R. Darade

Mobile: 8459693181

Email: daradem18@gmail.com

## Project Abstract

---

Google ads Hourly Analysis

This is Marketing Analysis for Education technology company.

In this data Impression indicates the visibility of the Advertisement, Clicks indicates the interested persons, Hot leads indicates that the number of person who are ready to purchase of service, Warm leads indicates the number of person who are really interested in service, Cold leads indicates the number of persons who are getting the information about services, CTR shows click through rate which is conversion rate of the customers, CPC shows cost per click. As we are looking for at which preferred time in a day where we can do marketing and we will get sales definitely.

We have analyze using Descriptive and Exploratory Analyses also further we have used Decision tree Classification.

# Google ads Hourly Analysis

## # Importing the libraries

```
In [2]: # import libraries
```

```
In [14]: import pandas as pd
import numpy as np
```

```
In [15]: data = pd.read_excel("C:/Users/91845/Downloads/Google ads hourly analysis 20th june.xlsx")
```

## # Importing the dataset

```
In [1]: # importing Data set
```

```
In [15]: data = pd.read_excel("C:/Users/91845/Downloads/Google ads hourly analysis 20th june.xlsx")
```

```
In [16]: data
```

```
Out[16]:
```

	Sr no	Impressions	Clicks	Cost	CTR	CPC	Cold Leads	Warm Leads	Hot Leads
0	00:00:00	9241	554	577	0.0600	1.041516	22	11	2
1	00:30:00	8873	355	498	0.0400	1.402817	14	7	1
2	01:00:00	3956	119	396	0.0300	3.327731	5	3	1
3	01:30:00	956	38	95	0.0400	2.500000	2	1	0
4	02:00:00	548	11	41	0.0200	3.727273	0	0	0
5	02:30:00	62	1	23	0.0200	23.000000	0	0	0
6	03:00:00	48	1	15	0.0200	15.000000	0	0	0
7	03:30:00	36	0	9	0.0120	NaN	0	0	0
8	04:00:00	19	0	7	0.0110	NaN	0	0	0
9	04:30:00	26	0	6	0.0190	NaN	0	0	0
10	05:00:00	45	1	7	0.0140	7.000000	0	0	0
11	05:30:00	59	1	12	0.0200	12.000000	0	0	0
12	06:00:00	36	0	9	0.0020	NaN	0	0	0
13	06:30:00	263	5	14	0.0200	2.800000	0	0	0
14	07:00:00	413	9	18	0.0220	2.000000	0	0	0
15	07:30:00	368	18	36	0.0500	2.000000	1	1	0
16	08:00:00	565	34	49	0.0600	1.441176	1	1	0
17	08:30:00	486	34	63	0.0700	1.852941	1	1	0
18	09:00:00	656	59	81	0.0900	1.372881	2	1	0
19	09:30:00	1025	123	148	0.1200	1.203252	5	3	1
20	10:00:00	1475	192	212	0.1300	1.104167	8	4	1
21	10:30:00	1488	85	180	0.0569	2.117647	3	2	0
22	11:00:00	1965	295	241	0.1500	0.816949	12	6	1

## # Processing Dataset

```
In [ ]: # Processing dataset
```

```
In [17]: a = data.drop("Sr no", axis=1)
```

```
In [18]: a.isnull().sum()
```

```
Out[18]: Impressions      0
Clicks      0
Cost      0
CTR      0
CPC      4
Cold Leads      0
Warm Leads      0
Hot Leads      0
dtype: int64
```

```
In [19]: b = a.fillna(a.mean() )
```

## # Splitting the dataset into the Training set and Test set

```
In [ ]: # Splitting the dataset into the Training set and Test set
```

```
In [25]: from sklearn.model_selection import train_test_split
```

```
In [117]: x_train, x_test, y_train, y_test = train_test_split( x,y, test_size=0.29, random_state=50)
```

## # Feature Scaling

```
In [118]: from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
```

```
In [119]: x_train = sc.fit_transform(x_train)
```

```
In [120]: x_test = sc.fit_transform(x_test)
```

## # Training the Decision Tree Classification model on the Training set

```
In [ ]: #Training the Decision Tree Classification model on the Training set
```

```
In [121]: from sklearn.tree import DecisionTreeClassifier
```

```
In [122]: DT = DecisionTreeClassifier(criterion="gini",random_state=0)
```

```
In [123]: DT.fit(x_train,y_train)
```

```
Out[123]: 

DecisionTreeClassifier  
DecisionTreeClassifier(random_state=0)


```

## # Predicting the Test set results

```
In [125]: from sklearn.metrics import confusion_matrix, accuracy_score
```

```
In [126]: y_prediction = DT.predict(x_test)
```

```
In [127]: y_prediction
```

```
Out[127]: array([4, 2, 5, 6, 0, 1, 1, 2, 1, 1, 0, 9, 0, 7], dtype=int64)
```

## # Prediction Accuracy

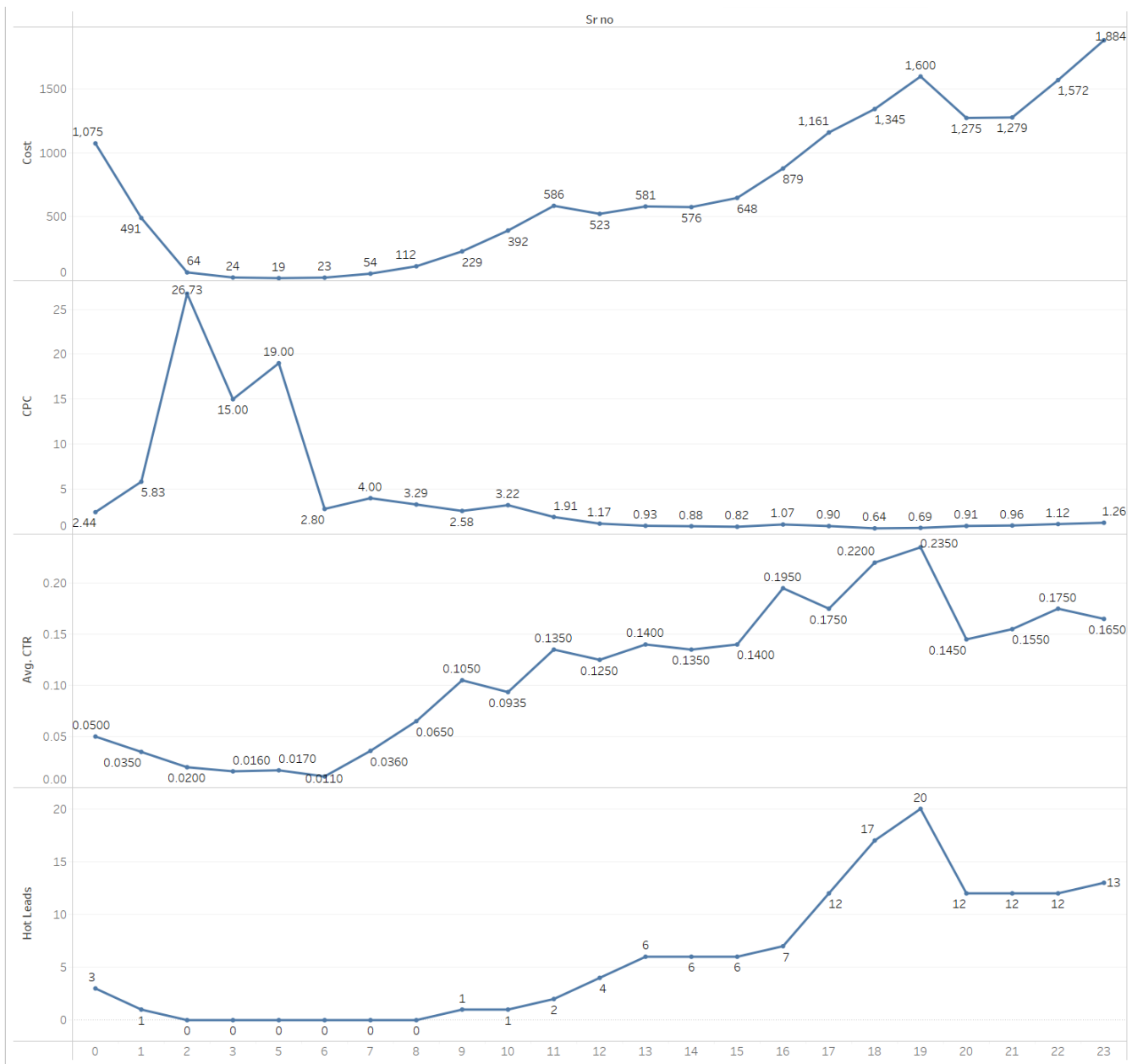
```
In [129]: accuracy_score(y_test,y_prediction)
```

```
Out[129]: 0.6428571428571429
```

## # Conclusion

From the given data we can conclude that the Decision tree is giving 64% accuracy for advertisement data.

## # Data Visualization (using Tableau)



## #insights

1. Company spend 16405 rupees on this day and got 135 hot leads.
2. Lead acquisition cost is 121.52 rupees.
3. After Analyzing this graph From 1am-9am average conversion rate is only 3% which is very low. In this time Cost per click also high which is 6.41 rupees and got very less Hot leads. Company can avoid this time for Marketing.
4. From 9:30am-5pm average conversion rate is 14%. In this time Cost per click is 0.73 rupee But 27% sell happen on this time.
5. From 5:30pm – 8pm Average conversion rate is high which is 21%. This time is very good for Marketing because of high conversion rate and low cost per click which is 0.36 rupee. In this time period 38% sell happen.
6. From 8:30pm-12:30pm 34% sell happen. Average conversion rate is 14% and cost per click is 0.69rupee. Company can continue their marketing in between this time