

## Website Performance & Engagement Analysis Using Python

Exploratory Data Analysis with Feature Engineering and Business Insights Project on Website Traffic, User Sessions, and Engagement Behavior

### 1. Introduction

In the digital era, understanding how users interact with a website is crucial for improving performance and user experience. This project focuses on analyzing website performance and user engagement using Python. The analysis covers traffic volume, engagement quality, session behavior, channel-wise performance, and time-based usage patterns.

The project is implemented entirely in Python using a Jupyter Notebook, making use of data cleaning, feature engineering, exploratory data analysis, and multiple visualizations to derive meaningful insights.

### 2. Project Objectives

- To analyze website traffic across different channel groups
- To understand user engagement using engagement rate and engaged sessions
- To compare engaged vs non-engaged sessions
- To analyze hourly traffic behavior
- To visualize trends and patterns using Python libraries
- To derive actionable business insights from data

### 3. Dataset Overview

The dataset used in this project contains website analytics data collected over time. Each record represents website activity for a specific channel at a given date and hour.

Key columns in the dataset include:

- Channel Group
- DateHour
- Users
- Sessions
- Engaged Sessions
- Engagement Rate
- Average Engagement Time per Session
- Events per Session

### 4. Tools and Technologies Used

- Python Programming Language
- Pandas: Data manipulation and analysis
- NumPy: Numerical operations
- Matplotlib: Basic plotting

- Seaborn: Advanced statistical visualization
- Jupyter Notebook: Development and analysis environment

## 5. Data Cleaning and Preparation

Before performing analysis, the dataset was cleaned and prepared:

- Column headers were corrected and standardized
- The first row of the dataset was removed after setting column names
- DateHour was converted to datetime format using pandas
- Numeric columns were safely converted using `pd.to_numeric`
- Missing and invalid values were handled using coercion
- A new feature 'Hour' was extracted from DateHour for hourly analysis

These steps ensured that the dataset was consistent, clean, and ready for analysis.

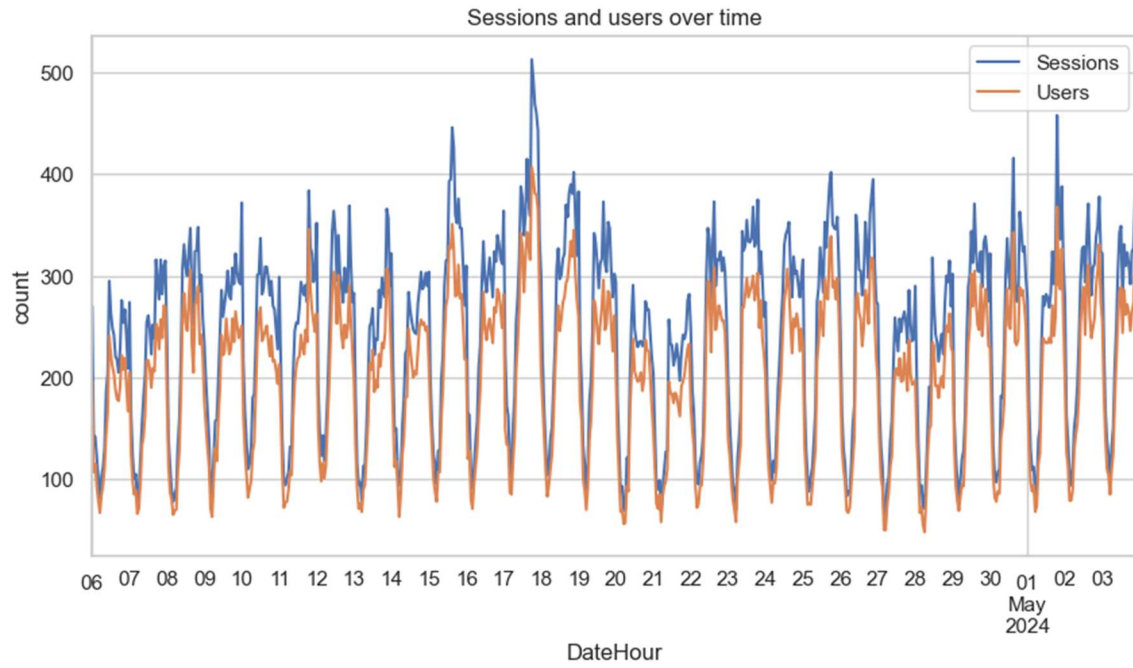
## 6. Exploratory Data Analysis (EDA)

Exploratory Data Analysis was performed to understand the structure and behavior of the data. Functions such as `df.info()` and `df.describe()` were used to inspect data types, missing values, and statistical distributions.

EDA helped identify variations in sessions, engagement rate, and user behavior across different channels and time periods.

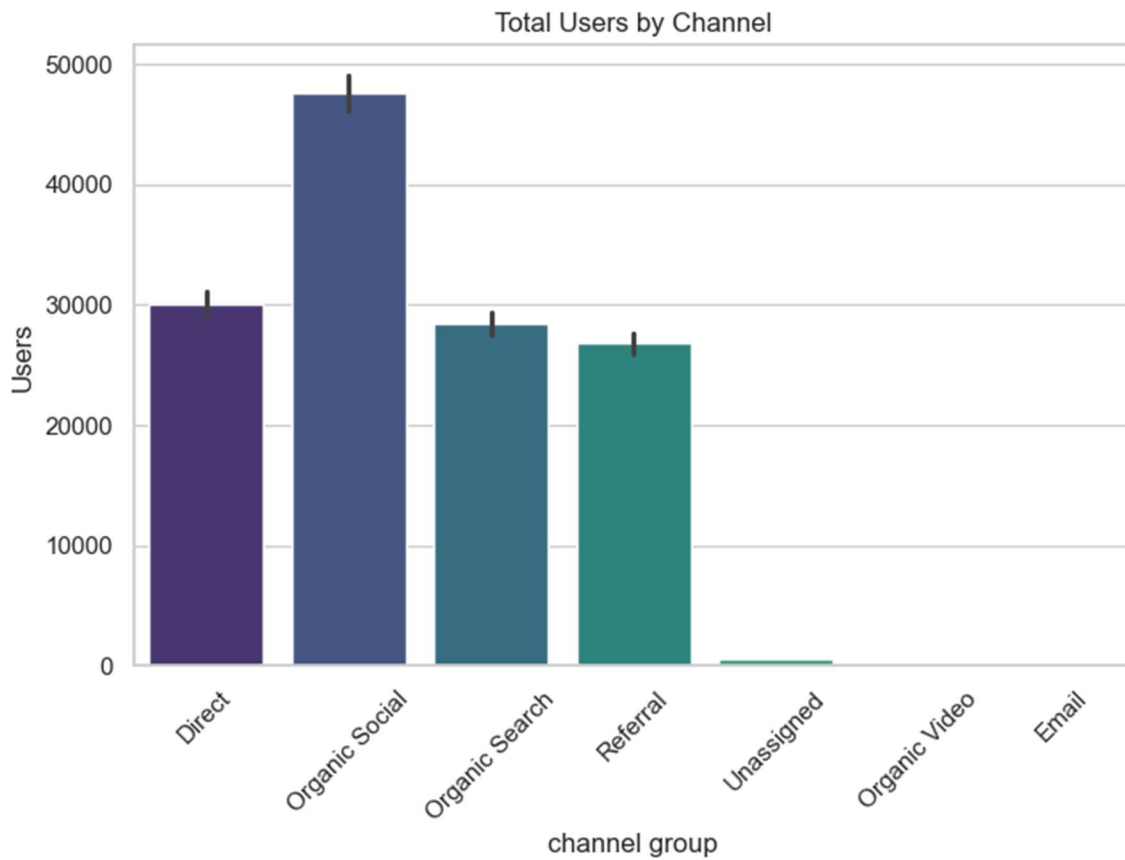
## 7. Visual Analysis and Interpretation

### 7.1 Sessions and Users Over Time



This line chart shows how sessions and users change over time. It helps identify traffic trends and peak periods.

## 7.2 Total Users by Channel



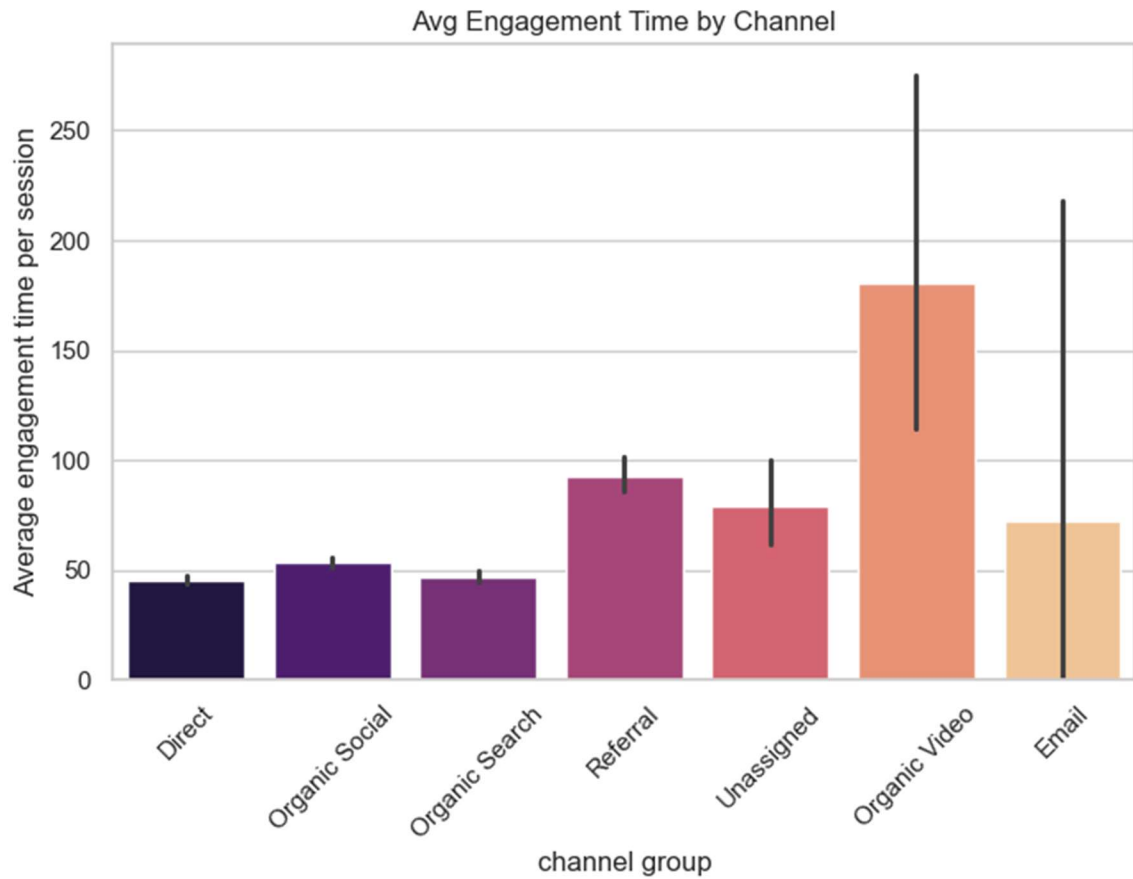
This chart answers:

Which channel brings maximum users

Which channel performs poorly

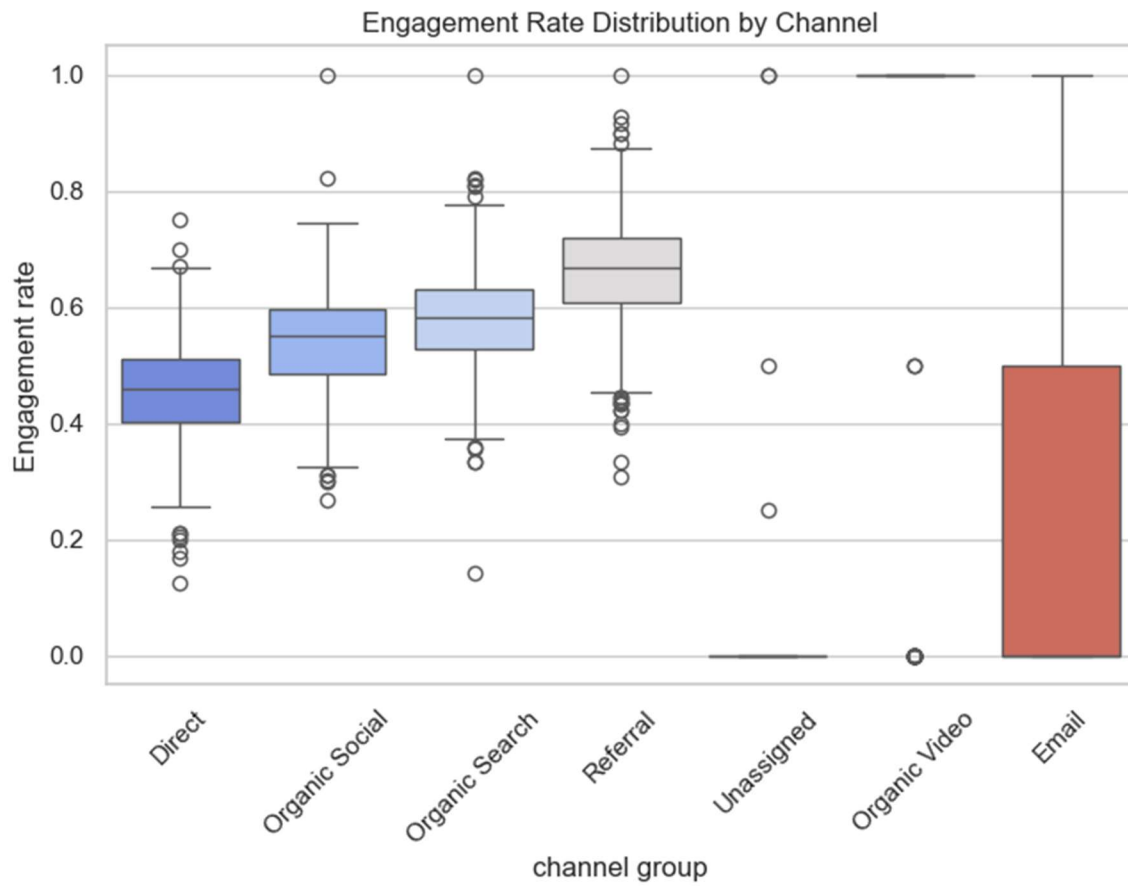
How user traffic is distributed

### 7.3 Average Engagement Time by Channel



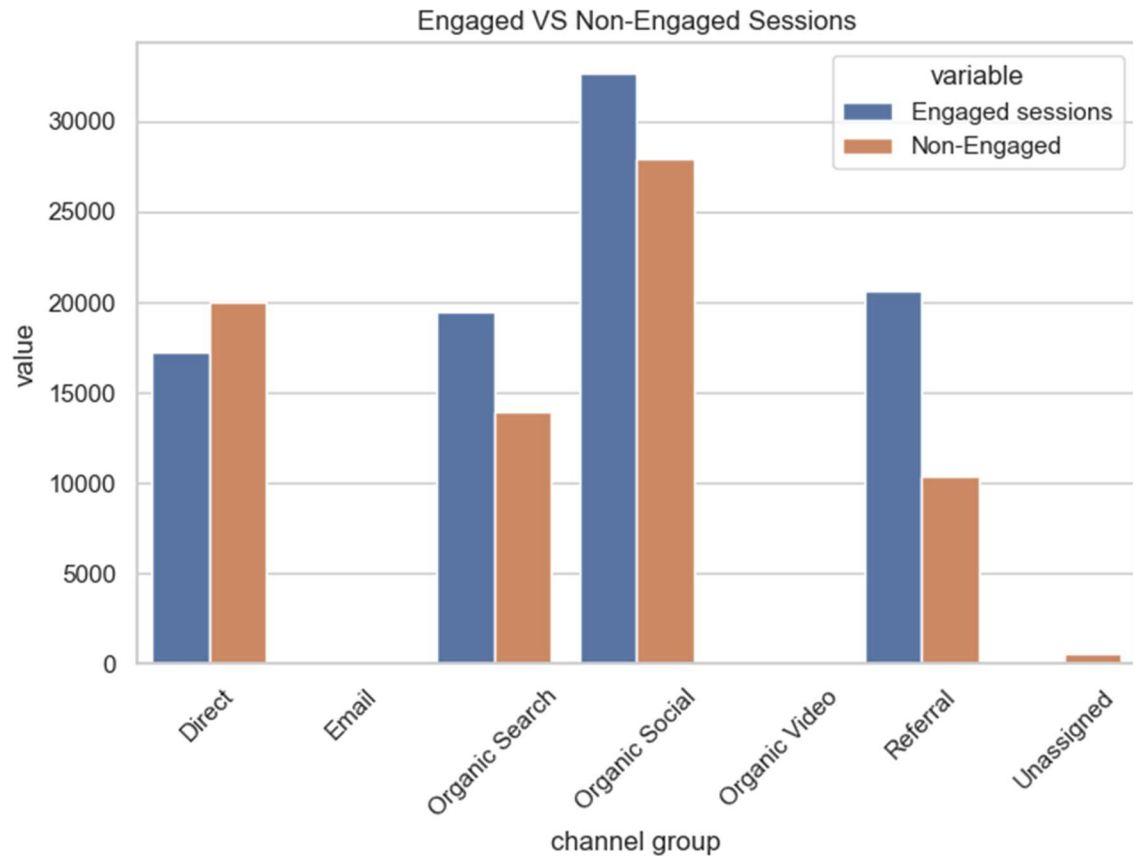
This bar chart compares the average engagement time per session across different channels. It highlights which channels bring more engaged users.

#### 7.4 Engagement Rate Distribution by Channel



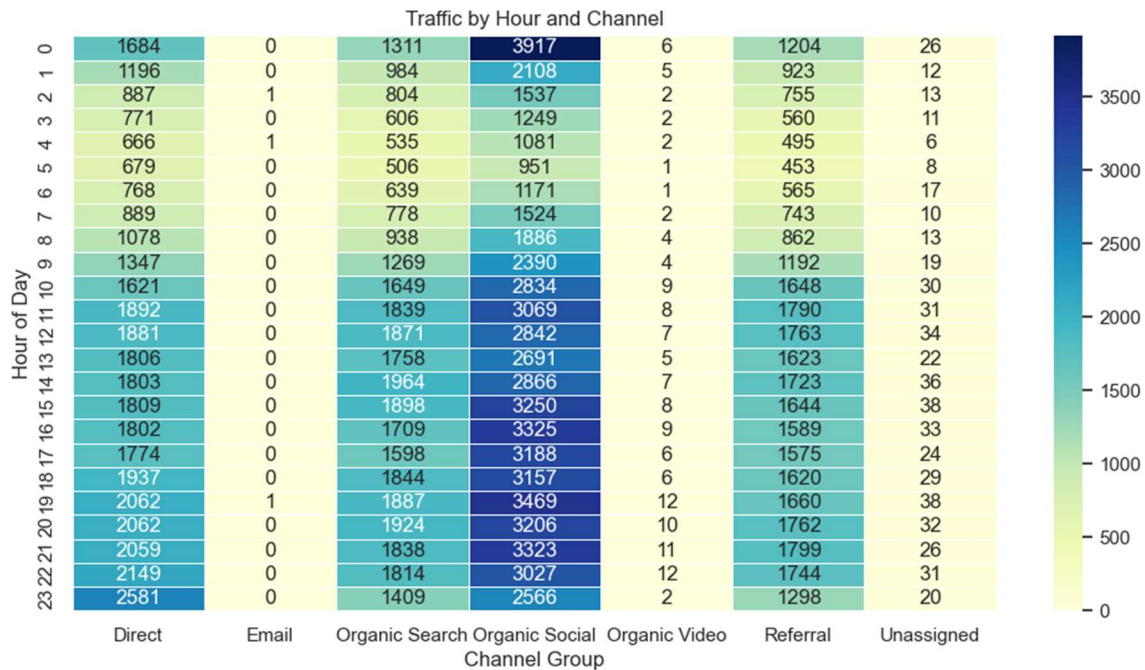
The box plot visualizes the distribution of engagement rate for each channel. Outliers indicate unusually high or low engagement sessions.

### 7.5 Engaged vs Non-Engaged Sessions



This bar chart compares engaged and non-engaged sessions across channels. It helps evaluate the quality of traffic from each channel.

## 7.6 Traffic by Hour and Channel



The heatmap shows session intensity by hour and channel. Darker colors represent higher traffic, helping identify peak activity hours.

## 8. Key Findings

- High traffic does not always lead to high engagement
- Some channels generate fewer sessions but better engagement quality
- Engagement behavior varies significantly by hour
- A considerable number of sessions are non-engaged
- Time-based and channel-based analysis reveals optimization opportunities

## 9. Business Insights and Recommendations

- Focus on channels with high engagement rather than only high traffic
- Improve landing pages for channels with high non-engaged sessions
- Schedule campaigns during peak engagement hours
- Optimize marketing strategies using data-driven insights

## 10. Conclusion

This project demonstrates the effective use of Python for website analytics. By combining data cleaning, feature engineering, exploratory analysis, and visualization, valuable insights were derived from raw website data.



The project showcases real-world data analysis skills and can be used for academic evaluation, professional portfolio, and interview discussions.