Multi Touch Attribution Use Case

Estimated Effort: 4 hours
Estimated Duration: 1 week
Programming Language: Python

Problem:

At SumUp, it is very important to understand and optimize our marketing efficiency. With customers interacting across multiple channels before making a purchase, we need to accurately assess the impact of each touchpoint in driving conversions. By doing so we aim at:

1. Analyzing the entire customer journey, from initial awareness to final conversion.
2. Assigning appropriate **credit** to each marketing touchpoint based on its influence on the purchase decision.
3. Gaining insights into the effectiveness of different channels and campaigns at various stages of the funnel.
4. Optimizing our marketing mix and budget allocation for maximum ROI.

Our current last-click attribution model fails to capture the full customer journey, potentially leading to misallocation of marketing resources and suboptimal campaign performance.

Credit refers to the number of conversions attributed to a marketing channel. In other words, it represents how many conversions a channel is believed to have contributed to.

Potential Solution:

Since last click attribution can be misleading and not very informative, we would like to implement a smarter approach. Multi-touch attribution assigns credit to marketing channels based on the entire **sequence** of user interactions, rather than just the last click.

Think of This as a Sequence Problem 🔄

User journeys follow a **pattern**, where each touchpoint influences conversion probability. Instead of treating interactions in isolation, consider how sequences evolve over time.

Hints for Your Approach 💡

- **Probabilities:** How does removing a touchpoint affect conversion likelihood?
- **Patterns:** Can you model user journeys to predict conversions?
- **Features:** What factors (order, timing, frequency) make a channel impactful?

- **Interaction Window:** How many days of interactions would you like to use to define a user journey?

Data:

We shared with you a dataset that will be useful in your analysis, that provides the following columns:

- user_id: A unique identifier for each user
- timestamp: Records the date and time of the user's touchpoint
- channel: Indicates the marketing channel through which the user interacted with our marketed product or brand (e.g., Organic Search, Paid Search, Social Media, Email)
- converted: A binary indicator (0 or 1) showing whether the user completed a purchase making them users of SumUp
- utm_medium: Identifies the marketing medium used, such as CPC (cost per click), organic, email, or social
- utm_source: Specifies the source of the traffic, like Google, Instagram,...
- utm_campaign: Denotes the specific marketing campaign associated with the user's interaction
- device_type: Categorizes the user's device, with possible values: desktop, mobile or tablet

Task:

1. Implement a last touchpoint attribution mechanism
    a. Analyse its output
    b. Briefly explain the disadvantages of such method
2. Build a sequence-based or a simple machine learning approach for multi touch attribution at **channel** level (the approach can involve Markov Chains, LSTM model, or any other approach you think is relevant)
    a. Analyse its output and compare it to last touchpoint attribution results
    b. Reuse the trained model to provide attribution results at any chosen level
    c. Choose an attribution level (e.g channel) and compare the outputs of this model to the previous approaches
3. [No code needed]:
    a. How would you assess which method provides the most reliable attribution results
    b. Based on the attribution results of your model, which recommendation can you share with the marketing department stakeholders? Think about the following:
        i. Which channels perform best in converting users at the end of the funnel
        ii. Which channels help in raising awareness at the beginning of the funnel
    c. Explain briefly the steps needed to take the machine learning model approach to production providing daily insights to stakeholders

You might need to carry out some data clean up or fix some data issues

Deliveries:

- You are supposed to ideally provide a jupyter notebook with your work on the questions

- Your notebook quality matters:
    - Try to keep your code readable and well commented
    - If you prefer, you can keep reusable parts of your code in separate python files
    - Use clear, descriptive markdown headers to structure your notebook into logical sections
    - Provide concise explanations of your analysis and findings in plain language
    - Focus on presenting results through easily digestible visualizations