Improving Starbucks' Promotional Decisions with Machine learning

[Capstone Proposal]

Domain Background

1

- Besides being the largest coffeehouse company in the world, Starbucks is notoriously known for its rewards program.
- So,When Starbucks launched its rewards program and mobile app, they dramatically increased the data they collected and could use to get to know their customers and extract info about purchasing habits.
- The data for this case simulates how people make purchasing decisions and how those decisions are influenced by promotional offers.
- Each person in the simulation has some hidden traits that influence their purchasing patterns and are associated with their observable traits.
- People produce various events, including receiving offers, opening offers, and making purchases.
- As a simplification, there are no explicit products to track. Only the amounts of each transaction or offer are recorded.
- There are three types of offers that can be sent:
    - Buy-one-get-one (BOGO), discount, and informational.
        - → BOGO: a user needs to spend a certain amount to get a reward equal to that threshold amount.
    - Discount: a user gains a reward equal to a fraction of the amount spent.
    - Informational: there is no reward, but neither is there a requisite amount that the user is expected to spend.

Offers can be delivered via multiple channels: Email, Mobile, Social, Web

Problem Statement 2

Not only does Starbucks go through mounds of coffee beans to satiate its raving fans, but they also have mounds of data that they leverage in many ways to improve the customer experience and their business.
Starbucks benefits since they would not be wasting money contacting the wrong customers for offers, or sending offers to customers who actually would have purchased anyway (or who are actually put off by receiving promotional 'spam').
Customers benefit, in that they receive offers that they are actually likely to be interested in and glad to receive, rather than just a blanket sending of offers that are not relevant to many customers.
We will use the data to find a better promotion strategy by Identify which groups of people are most responsive to each type of offer, How best to present each type of offer?, How many people across different categories actually completed the transaction in the offer window? We will also try to train a model to predict the amount that can be spent by an individual given the individual's traits and offer details. This will help us decide which promotional offer best suits the individual and respond to the target audience with better accuracy.

Datasets and Inputs

3

Udacity and Starbucks as part of the Machine Learning Engineer Nanodegree program and provided source in this [link](). It contains simulated data that mimics customer behavior on the Starbucks rewards mobile app.
The program used to create the data simulates how people make purchasing decisions and how those decisions are influenced by promotional offers.Each person in the simulation has some hidden traits that influence their purchasing patterns and are associated with their observable traits. People produce various events, including receiving offers, opening offers, and making purchases.

<u>Datasets provided:</u>
Three json files were provided with simulated data:

- portfolio.json :
  - This is a very short file giving details of the 10 unique offer ids used by Starbucks in this experiment. It provides details including reward (the dollar level of discount), difficulty (the dollar spend level required to trigger the offer), duration (the number of days for which the offer is valid),the channels (how the offer was sent to the customer) and the offer type

Offers sent during 30-day test period (10 offers x 6 fields)

- ***reward***: (numeric) money awarded for the amount spent
- ***channels***: (list) web, email, mobile, social
- ***difficulty***: (numeric) money required to be spent to receive reward
- ***duration***: (numeric) time for offer to be open, in days
- ***offer_type***:(string) bogo, discount, informational
- ***id***: (string/hash)

- profile.json :
  - This is a dataset containing 17,000 members, including demographic details such as gender, age, income and the date they became a member of the loyalty programme.

Rewards program users (17000 users x 5 fields)

- ***gender***: (categorical) M, F, O, or null
- ***age***: (numeric) missing value encoded as 118
- ***id***: (string/hash)
- ***became_member_on:*** (date) format YYYYMMDD
- ***income***: (numeric)

Datasets and Inputs 3

- **transcript.json:**
  - **This is a large dataset containing over 300,000 rows of event data for four different event types: "offer received", "offer viewed", "offer completed" and "transaction"**

Event log (306648 events x 4 fields)

- ***person***: (string/hash)
- ***event***: (string) offer received, offer viewed, transaction, offer completed
- ***value***: (dictionary) different values depending on event type
  - *offer id*: (string/hash) not associated with any "transaction"
  - *amount*: (numeric) money spent in "transaction"
  - *reward*: (numeric) money gained from "offer completed"
- ***time***: (numeric) hours after start of test

Soultion Statement 4

The data will be carefully analyzed so the business understanding can be fine-grained, and insights can be generated. Then, we will proceed to feature engineering, where useful features will be created from variables.
In order to do so, it will be necessary to concatenate and join the datasets properly.
Then, the problem will be developed according to a classification framework - that is, features and labels will be clearly defined.
Since the problem is complex by design, we will use more advanced algorithms.
The only care we need to take here is avoid overfitting, since such models can adjust too well to training data.
That is why it is important to split the dataset accordingly - avoiding same pair user-offer appear in evaluation and training, which could lead to data leakage. By doing so and comparing metrics between training and validation sets, we are able to take overfitting into account.
Finally, an algorithm will be tuned in order to find the best possible performance. The final evaluation will be held on a dataset which was not previously used during model design.
Finally, this model will be made available through an API, so end users are able to perform requests to an endpoint to get predictions - probabilities to a given scenario. In order to accomplish it, training and deployment will be carried out in AWS SageMaker with integration with AWS Lambda.

Benchmark Model 5

AI and Machine Learning are on track to generate Trillion in value by solving Marketing and Sales problems over the next three years.
The use of AI soared between 2018 and 2020, jumping from 29% in 2018 to 84% in 2020, according to Salesforce Research's Study.

70% of high-performance of ML teams claim they have a fully defined AI strategy versus 35% of their under-performing peer marketing team counterparts. CMOs who lead high-performance marketing teams place a high value on continually learning and embracing a growth mindset, as evidenced by 56% of them planning to use AI and machine learning over the next year. Choosing to put in the work needed to develop new AI and machine learning skills pays off with improved social marketing performance and greater precision with marketing analytics.

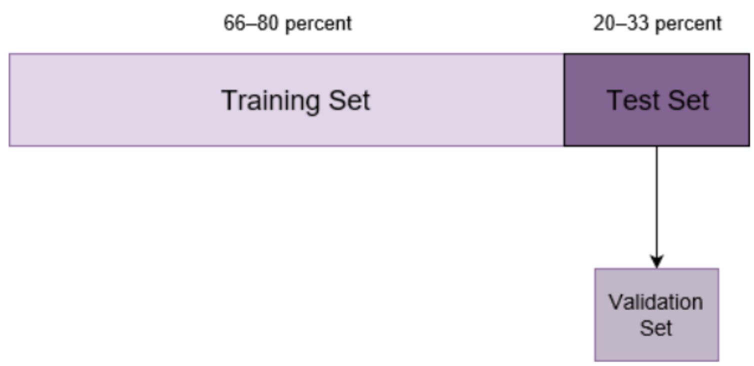
Evaluation Metric 6

Evaluation metrics are used to measure the quality of the model. One of the most important topics in machine learning is how to evaluate your model. When you build your model, it is very crucial to measure how accurately it predicts your expected outcome.

Evaluation metrics can help you assess your model's performance, monitor your ML system in production, and control your model to fit your business needs.
Our goal is to create and select a model which gives high accuracy on out-of-sample data. It's very crucial to use multiple evaluation metrics to evaluate your model because a model may perform well using one measurement from one evaluation metric while may perform poorly using another measurement from another evaluation metric.
Classification Accuracy such as recall,precision, accuracy and ROC AUC.
The simplest metric for model evaluation is Accuracy. It is the ratio of the number of correct predictions to the total number of predictions made for a dataset.
Understanding how well a machine learning model will perform on unseen data is the main purpose behind working with these evaluation metrics. Metrics like accuracy, precision, recall are good ways to evaluate classification models for balanced datasets, but if the data is imbalanced then other methods like ROC/AUC perform better in evaluating the model performance.
ROC curve isn't just a single number but it's a whole curve that provides nuanced details about the behavior of the classifier. It is also hard to quickly compare many ROC curves to each other.

Project Design 7

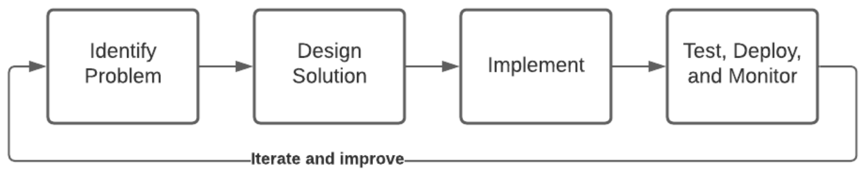
- Strategy: matching the problem with the solution
    - In the first phase of an ML project realization, a solution to a problem, define a scope of work, and plan the development.
- Dataset preparation and preprocessing.
    - Data is the foundation for any machine learning project. The second stage of project implementation is complex and involves data collection, selection, preprocessing, and transformation.
- Dataset splitting
    - A dataset used for machine learning should be partitioned into three subsets — training, test, and validation sets.



- Modeling
    - we preprocessed the collected data and split it into three subsets,then we can proceed with a model training. This process entails "feeding" the algorithm with training data. An algorithm will process data and output a model that is able to find a target value (attribute) in new data.

- Model evaluation and testing
    - The goal of this step is to develop the model to be able to formulate a target value fast and well enough. As we can achieve this goal through model tuning. That's the optimization of model parameters to achieve an algorithm best performance.

- **Model deployment**
    - **The model deployment stage covers putting a model into production use.**



AWS Solution Scheme