MSDS 629: Final Project

DUE: Tuesday January 21, 2025 by 9:00am PST

PREAMBLE

Streaming services are well known for their use of controlled experiments to innovate and iterate on their platforms. Netflix, in particular, has a strong culture of experimentation. This culture and the "wins" that experiments have brought about at Netflix, have made it widely renowned as an industry leader in online experimentation. Blog posts like this 7-part series about experimentation at Netflix (Part 1, Part 2, Part 3, Part 4, Part 5, Part 6, Part 7) and jobs ads like this one that explicitly mention multivariate testing and response surface methodology, illustrate the company's experimental maturity. In this project you will embark on a Netflix-inspired experimental journey in which you conduct experiments for a hypothetical problem at a hypothetical competitor called *Webflix*.

THE PROBLEM

In this project you will be concerned with optimizing the Webflix homepage by way of minimizing *browsing time*. For those unfamiliar with Webflix, a screenshot of the homepage is included above. As is depicted in the screenshot, the homepage is laid out in a grid system in which movies and TV shows appear as tiles with rows differing with respect to some categorization. Though not depicted in the screenshot, when one hovers their mouse over a tile, its size is enlarged and a preview of the show/movie is automatically played in the enlarged window.

When faced with so many viewing options, Webflix users often experience choice-overload and can be overcome by a psychological phenomenon known as decision paralysis. The problem is that it becomes harder to make a decision, and it takes longer to make a decision, when faced with a large number of options to choose from. Decision paralysis negatively impacts Webflix because a user may become overwhelmed by all of the options and fatigued by the prospect of making a choice, and may ultimately lose interest and not watch anything.

To overcome this, Webflix tries to help you choose what to watch, and by a variety of mechanisms tries to help you choose quickly. Of relevance is browsing time – the length of time a user spends browsing (as opposed to watching) Webflix. Ideally, browsing time and, in particular, average browsing time would be small. In this project you will conduct a series of experiments to learn *what* influences browsing time and *how* that may be exploited in order to minimize average browsing time. There are infinitely many things that likely influence the amount of time someone spends browsing Webflix, but just four factors will be explored in this project. Each is related to the "Today's Top Picks for You" row of the Webflix homepage. This row contains recommendations algorithmically curated for the specific user.

- **Tile Size:** The ratio of a tile's height to the overall screen height. Note the tile's aspect ratio is fixed so changing this factor changes the size of the tile, but not its shape. Smaller values correspond to a larger number of tiles visible on the screen, and larger values correspond to fewer visible tiles.
- **Match Score:** A prediction of how much you will enjoy watching the show or movie, based on your viewing history. This is recorded as a percentage, with larger values indicating a higher likelihood of enjoyment.
- **Preview Length:** The duration (in seconds) of a show or movie's preview.
- **Preview Type:** The type of preview that is autoplayed.

The table below summarizes the region of operability for each of these factors, and the default values they take on when not being experimented with.

Factor	Code Name	Region of Operability	Default Value
Tile Size	`Tile.Size`	[0.1,0.5]	0.2
Match Score	`Match.Score`	[0,100][a]	95
Preview Length	`Prev.Length`	[30, 120][b]	75
Preview Type	`Prev.Type`	{`TT`, `AC`}[c]	`TT`

[a] For purposes of experimentation `Match.Score` must be an integer. You may assume that all titles recommended in the "Today's Top Picks for You" row will have a match score greater than or equal to `Match.Score`.

[b] For purposes of experimentation `Prev.Length` can only be changed in increments of 5 seconds.

[c] `TT` stands for *teaser/trailer* and `AC` stands for *actual content*.

Through a series of experiments you will seek to determine which of these factors significantly influences browsing time, and you will attempt to find an optimal configuration of them that minimizes expected browsing time. You will do this by interacting with a web-based simulator, into which you will submit experimental designs and out of which you will receive response observations.

The remainder of this document provides guidelines for using the simulator, an overview of the experimentation process you will undertake, and a description of the deliverable that you must submit. An outline of the marking scheme is also included to make clear my expectations and to make transparent the manner in which you will be graded.

THE SIMULATOR

The response surface simulator can be accessed at the following URL:

https://nathaniel-t-stevens.shinyapps.io/Webflix_Simulator/

The interface and the manner in which you interact with it is straightforward: you upload a design matrix and then collect your results. Interaction with the simulator should include three distinct steps:

1. Upload a `.csv` file containing your design matrix. The `.csv` file **must** adhere to the following formatting guidelines:

    - The file name must be your unique 8-character group code, (e.g., `NSTEVENS.csv`). Any file name other than this will result in an error.
    - The columns must have the headings `Tile.Size`, `Match.Score`, `Prev.Length`, and `Prev.Type`, corresponding to the design factors. The order of the headings does not matter and you do not need to experiment with every factor in every experiment, in which case not all columns (and hence headings) are required.
    - Each row corresponds to a distinct experimental condition, and each element indicates the level of the corresponding factor.
    - There should be a final column called `Sample.Size` in which you specify the number of experimental units to assign to each experimental condition.

2. Click the "Visualize my Design" button. This will render one or more plot of the design space that indicate the experimental conditions you plan to run.

    - If the design is not the one you intended, you may reset the simulator (by clicking the "Reset" button) and upload a different design matrix.
    - If there is anything amiss with the file you uploaded, an error (instead of a plot) will be returned.
    - **IMPORTANT:** Make sure to click the "Reset" button prior to every subsequent `.csv` upload.

3. Supposing you are happy with the design, click the "Run the Experiment" button. This will generate browsing times (recorded in minutes) for each unit in each condition you specified. The results will be automatically downloaded in a `.csv` file.

    - Remark 1: This mimics the random assignment of users to each condition and the observation of their response variable.
    - Remark 2: You may assume that browsing time observations do not include the amount of time spent watching previews; browsing time records only the time spent scrolling and searching.

THE EXPERIMENTS

Your experimental journey will consist of a series of experiments in which you investigate and optimize the relationship between browsing time and `Tile.Size`, `Match.Score`, `Prev.Length`, and `Prev.Type`. Subject to the constraints listed below the table on page 2, you will have complete control over the choices you make throughout this journey. Which factors to experiment with, which levels to choose, and which conditions to explore are all up to you. This mimics the real world, where problem solving is typically unstructured and often non-linear.

At the conclusion of your journey you should have identified the location of the optimum. That is, the combination of values for `Tile.Size`, `Match.Score`, `Prev.Length`, and `Prev.Type` that you believe minimize browsing time. Report the estimate and a 95% confidence interval for the expected browsing time at this location.

THE DELIVERABLE

In groups of three students, you will perform the experiments described above and prepare and submit a report via Crowdmark by the due date listed at the top of this document.[1] The report will document your experimental journey and will consist of the following components (each of which must begin on a separate page):

- Component #1: Executive Summary (4 sentences max)
  - Summary of the problem, your experimental journey, and the ensuing findings.
  - Be sure to state the location and value of the optimum.
- Component #2: Introduction (0.5 page max)
  - Describe *in your own words* the problem you are trying to solve.
  - Describe *in your own words* the experimentation methods you chose to employ.
  - Provide an outline of what the reader will find in the remainder of the report.
- Component #3: The Experiments (3 pages max)
  - Describe your experimental journey.
  - Explain your experiments through the lens of QPDAC. State their objective, explain their design, collect the data, analyze the data, and draw a conclusion.
  - Be sure to *justify any decisions you made* in either the design or the analysis of these experiments and make clear how learnings from one experiment influenced decisions made in future experiments.
  - Be sure to include visual and/or tabular summaries that are relevant to the experiment.
- Component #4: Conclusion (0.5 page max)
  - State the conclusion of your experimental journey. That is, the location and value of the optimum.
  - Acknowledge any limitations that might be associated with your findings.

IMPORTANT: Your report *will not contain* Python code or Python output. Discussion of your analyses should be succinct, and analysis results should be included as figures and/or nicely formatted tables. Note that figures and tables count toward the page limit. Include only that which is necessary to tell your story and to justify your decisions.

[1] Note that only one member of your group needs to submit the report.

The MARKING SCHEME [40 points]

REPORT [30 points]

- Executive Summary [4 points]
  - [2] Grammar, professionalism
  - [2] Clarity, brevity
- Introduction [7 points]
  - [2] Grammar, professionalism
  - [2] Clarity of problem recapitulation
  - [3] Clarity, coverage/depth, relevance of methodology overview
- The Experiments [15 points]
  - [2] Grammar, professionalism
  - [2] Clarity of objectives
  - [4] Suitability of design and clarity of design choices
  - [4] Suitability of analysis and clarity of analysis choices
  - [3] Suitability and clarity of conclusions
- Conclusion [4 points]
  - [2] Grammar, professionalism
  - [2] Clarity, relevance

ACCURACY & EFFICIENCY [5 points]

- Accuracy of Optimum [3 points]
  - [3] The optimum you've identified is **very close** to the true optimum.
  - [2] The optimum you've identified is **close** to the true optimum.
  - [1] The optimum you've identified is **somewhat close** to the true optimum.
  - [0] The optimum you've identified is **not at all close** to the true optimum.
- Efficiency of Experimentation[2] [2 points]
  - [2] If the total number of experimental units is $\leq 25,000$
  - [1] If the total number of experimental units is $> 25,000$ and $\leq 50,000$
  - [0] If the total number of experimental units is $> 50,000$

[2] If you want to play around with the simulator without sacrificing your efficiency score, I recommend playing with the `NSTEVENS` account, but note that it has a different underlying response surface than yours. Exploring it will not provide any insight for your surface.

TEAMWORK [5 points]

To a *personal* submission folder on Crowdmark, upload a team member evaluation document that contains the following information: (1) your 8-character team code, (2) the names of your team members (including yourself), and (3) beside each team member's name (including yourself), a number between 0 and 1 that indicates the proportion of work completed by that member. These values must sum to 1.

I will then use these evaluations to calculate your "Teamwork" score as follows. Let $\{x_1, x_2, x_3\}$ be the three peer evaluations for person i (including their self evaluation). Define $m_i = \text{median}\{x_1, x_2, x_3\}$ to be the median evaluation for person i. The "Teamwork" score for person i is then calculated as

$$5 \times \frac{m_i}{\max\{m_1, m_2, m_3\}}.$$

Please submit your peer evaluations by the due date listed at the top of this document (i.e., the same date and time the final report is due).