Team 20 - Plutionians - Marketing Campaign Analysis

Calvin Raab, Chris Struck, Justin Barry, Naz Uremek

Introduction

In today's digital era, marketing is continuously evolving in data availability and technology. From ad hoc reporting insights to AI-powered machine learning methods, data driven decisions have become an integral part of marketing in various industries. Two popular themes include campaign effectiveness and forecasting. Marketing teams are spearheading the use of machine learning with evaluating campaign results to understand how best to engage with their customers. Fine tuning the campaign process by targeting groups of individuals who are more likely to install or subscribe to a product allows for companies to allocate their marketing resources and budget more efficiently.

This project aims to satisfy PlutoTV's marketers' needs in two ways: (1) to present a one-stop shop tool to analyze campaigns visually, gaining high level insights, and (2) to plan future campaigns using various machine and deep learning techniques such as random forest, k-means, and Long Short Term Memory (LSTM).

The main business objectives include determining:

1. What is the impact of campaign spend from this week to next week for total active users, viewed minutes, and on minutes watched per user?
2. For campaign budgeting optimization for upcoming quarters, how can marketing best spend their budget to optimize spend across marketing partners to increase ROI?
3. Which states bring in the most users through campaign ads? Which states have the most total minutes watched per user from an ad?

Related work

Marketing research covers a spectrum from applied methods like A/B testing to theoretical discussions on the strategic value of marketing. O'Sullivan and Abela define Marketing Performance Measurement (MPM) as a firm's capacity to evaluate how marketing efforts drive business outcomes. They argue that strong MPM systems are essential for justifying marketing expenses and building credibility within the organization (O'Sullivan and Abela, 2007). Firms with robust MPM capabilities show superior financial results, such as higher profitability and sales growth. The study emphasizes that MPM helps marketing leaders build trust and influences strategic decisions. Ultimately, investing in MPM systems is crucial for maximizing the impact of marketing and demonstrating its value to executives, making it potentially valuable for PlutoTV.

Clemens Koob's research explores the factors contributing to the success of content marketing campaigns, along with strategies to increase their effectiveness (Koob, 2021). One key finding is that neither the number of media platforms used nor the specific distribution channels significantly influence campaign outcomes. Instead, the study emphasizes that the quality and alignment of content with the audience's interests are the primary drivers of success. This insight is valuable for informing the design of our dashboard, as it suggests that our recommendations should focus on optimizing content quality rather than simply suggesting an expansion of media distribution.

Pursuant to our mentor's feedback of emphasizing prior research to inform our approach, this study and the following study's results guided us to prioritize the analysis of genre with respect to campaigns and the states where they were distributed, helping to contextualize business objective 3 with respect to our available data. This ultimately affected our EDA process as well as our final dashboard and data cleaning process. Similar to Koob's research, Anita Lopes and Beatriz Casais emphasize that aligning content with audience needs is essential for success. The authors introduce a strategic framework that categorizes companies as "emerging," "developing," or "maturing" based on their content marketing practices (Lopez and Casais, 2022). We integrated a similar classification process into our tool to assess and predict a campaign's results on four levels based on the number of users compared to other campaigns. Similar to Koob's paper, Lopes and Casais emphasize that aligning content with the audience's needs is essential for success, so we focused the predictive model on genre. As we will discuss later, we found that genre had very little effect across states, but a larger impact on campaign success.

Additional previous research discussed the results of integrating various classifiers, regression, and deep learning methodologies. Mainak Sarkar and Arnaud De Bruyn's research uses direct marketing to observe product purchases before campaign exposure, and after campaign exposure. The results claim that LSTM models taking raw behavioral data as input with little to no feature engineering achieve a better average fit and performance than the feature-based, benchmark models (Sarkar and Bruyn, 2020). Because of this, we decided to study the results using both numerous engineered features and the core group of KPIs inherent in the dataset, focusing on investigating clicks and impressions results on a campaign to campaign basis.

Methodology

We addressed the target business problem by developing a Streamlit dashboard equipped with built-in machine learning tools. The dashboard includes a campaign analyzer that features time series analysis, key performance metrics, and geographic data processing. Additionally, there is a dedicated section for ML-based predictions, providing a seamless integration of analysis and predictive capabilities within the application.

Data Preparation

To measure the success of Pluto TV campaigns, we implemented significant data engineering in order to structure the dataset properly for both dashboard visualizations and machine learning. Our team primarily used two levels of aggregation, both of which are used interchangeably for visuals as well as the machine learning methods.

Device level: The Pluto TV data was initially at the impression level, meaning each device had multiple rows for each impression. This caused bias in modeling and visualizations, especially for linear models, as some devices appeared thousands of times while others appeared only a few. To resolve this, we aggregated the data to the device ID level, summing relevant columns and standardizing categories (e.g., taking the first value for repeated attributes like creative names). One-hot encoding was added to prepare the data for modeling. The processed device-level dataset became the foundation for further steps and was automated through the `processing_module.py`.

Campaign level: Most of our modeling was focused on the campaign level rather than individual devices. To achieve this, we aggregated the device-level data by campaign, summing variables and removing outliers. Campaigns with fewer than 30 devices were excluded to maintain statistical reliability. We created a dependent variable, "score," by categorizing average watch time per device into quartiles: Poor, Fair, Good, and Excellent. This provided a clean, structured dataset ready for modeling.

State and Genre Encoding: To add geographical context, we mapped latitude and longitude coordinates to state abbreviations (e.g., California → CA) using a YAML file. This approach bypassed API limitations and was efficient for state-level analysis. While finer geospatial details were omitted, this method enabled basic spatial insights. Future iterations can expand on this by incorporating more granular geospatial data.

Anonymization: To ensure data security, sensitive information was anonymized through an ETL function. The function replaced sensitive values with a standardized format (e.g., column name + numeric ID). A mapping of original-to-anonymized values was saved in a "mappings" folder for retrieval by authorized users under an NDA. This allowed the creation of a protected dataset ready for release while maintaining traceability of sensitive information.

Feature Engineering: One-Hot Encoding was used to convert categorical variables into a series of binary columns. Similarly, for numerical columns, **standard Scaler** standardized features by removing the mean and scaling to the unit variance. Lastly, **binning** was implemented to recategorize values in groupings to reduce the count of distinct values within a feature.

Machine Learning Methods

Unsupervised Learning

K-means: The team used various clustering methods both to analyze the data and as a potential predictive tool in sifting through a variety of outcomes. We believed that analyzing watch time across a variety of campaigns and genres was a well suited method given the disparate nature of the target business objectives. We initially used this method to explore genre and state data, which informed additional cleaning and transformation steps.

Nearest Neighbor: We implemented a nearest neighbor model to identify the most similar campaign from the training data based on Euclidean distance. This allows users to compare the performance of a similar past campaign to the one they just created, providing deeper insights. The model was configured to return only the single closest campaign.

Supervised Learning

Decision Tree Classifier: We built a feature to predict the score (Poor, Fair, Good, Excellent) of a user-created campaign based on clicks, impressions, and genre percentages using a Random Forest Classifier.

Decision Tree Regressor: We aimed to predict the total average minutes watched per device as a measure of campaign success—longer watch times indicate better campaigns. To achieve this, we built a decision tree regressor.

Deep Learning

Long Short Term Memory (LSTM): Combining unsupervised and supervised learning methods with a neural network helped us better understand predictions over time. LSTM uses long-term dependencies in sequential data, which can be applied to our time series dataset. The following steps were taken in order to properly run the model:

1. Preprocessing
2. Preparing sequences
3. Building LSTM model
4. Plotting training history
5. Plotting actuals and predictions

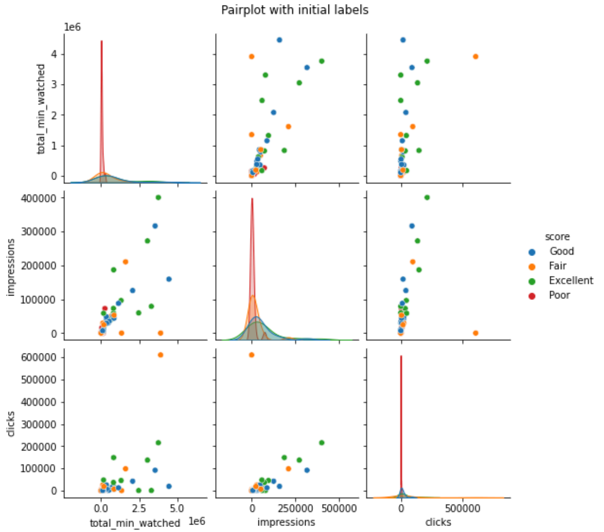
Results

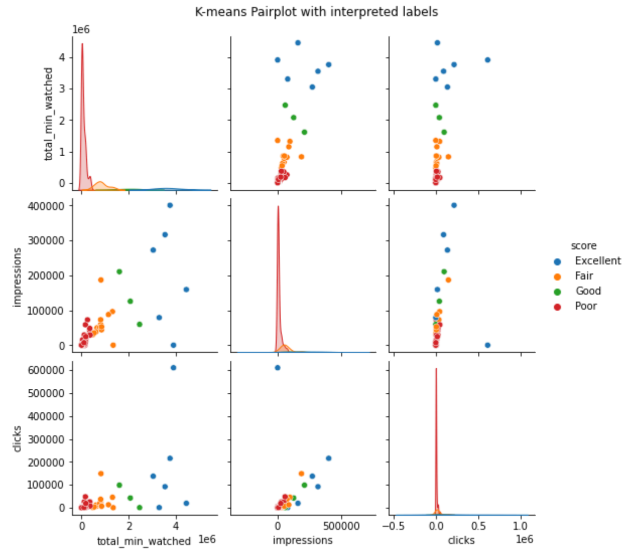
By using the aforementioned blend of machine learning methods, including supervised and unsupervised processes, we hoped to provide a valuable dashboard tool directly to senior management at PlutoTV that helps to evaluate past campaigns while informing new ones. One challenge that we encountered was determining the right metrics to evaluate a campaign's success. Per the research that we discussed earlier, our focus was on a campaign's impact, so we prioritized Average Minutes Per User and the speed at which PlutoTV was downloaded after an advertisement over metrics such as total minutes watched across a campaign. This created unique challenges with finding the best machine learning algorithms.

Machine Learning

Unsupervised Learning

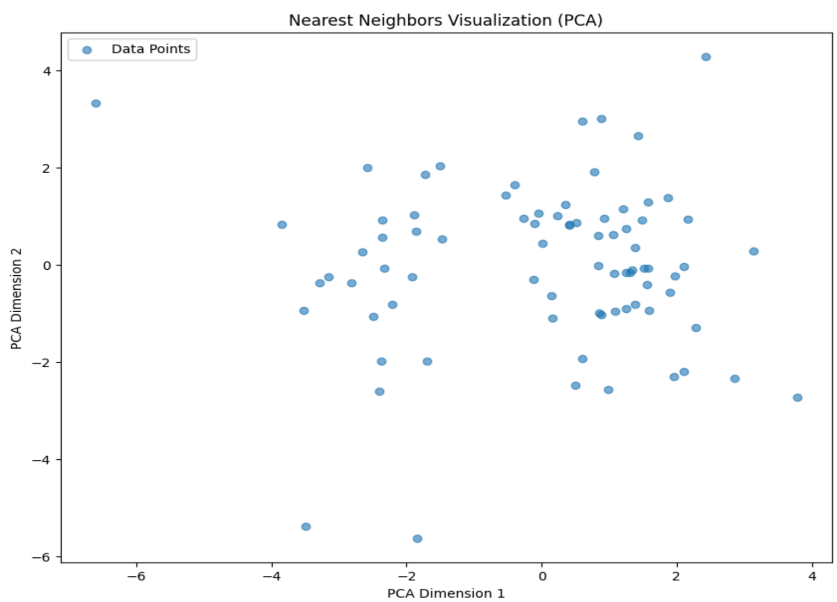
K-means: With the final campaign level, we attempted to utilize K-means alongside KNN to decipher the linearity and predictability in our data. We found that KNN had a 45% predictive accuracy on the aggregate campaigns, and when passing the same data through K-means, we found that there was a 34% overlap in labels between our initial label and the predicted label. Please note that a highly predictive dataset that scales linearly on volume will have a near 1:1 ratio between K-means and KNN results. The following pair-plots show the differences between KNN (left) and K-means (right):





Nearest Neighbors:

The nearest neighbor algorithm is unsupervised, so accuracy isn't directly measurable. To explore the data, we used PCA to reduce the features to two dimensions for visualization. The plot shows a general division into two regions, but distinct neighborhoods or clear patterns are hard to identify.



Supervised Learning

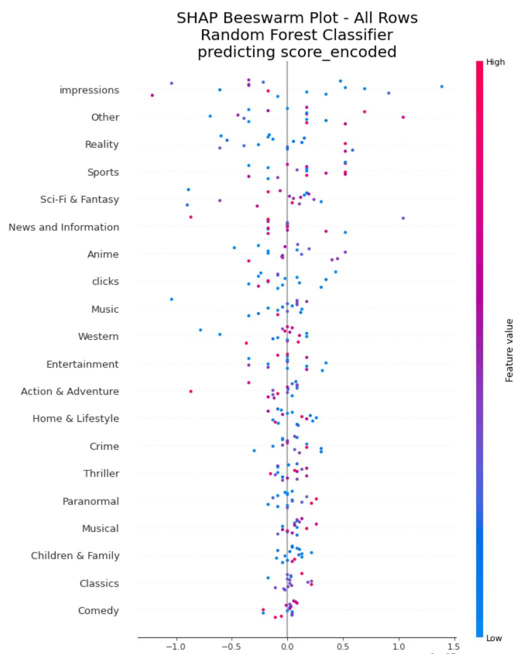
We decided to run prediction models based on two features selected: *Ad Impression to Watch Time within 7 Days* and *Average Minutes per User*.

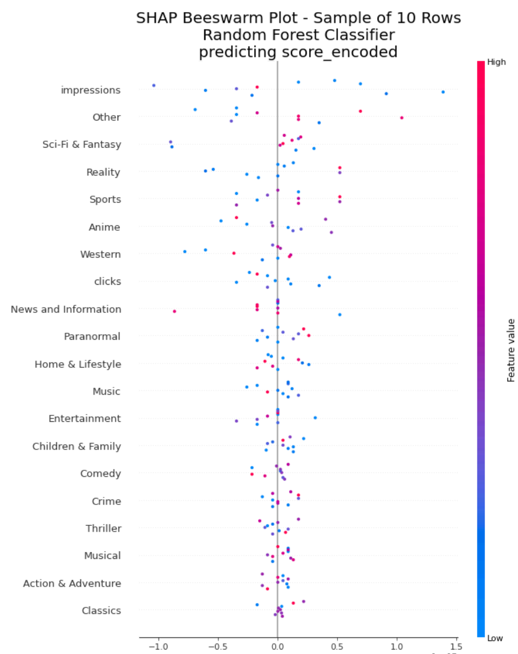
Ad Impression to Watch Time within 7 Days

Boolean indicator *imp_event_7days* categorizes whether or not a user watched content within seven days of seeing an ad. Comparing four models in order to determine each model's success, four metrics were measured: accuracy, recall, precision, and f1 score. Logistic Regression took the lead for precision (74%), Random Forest Classifier for accuracy (68%), and Gradient Boost Classifier for recall (92%) and f1 score (77%). Decision Tree Classifier resulted in the lowest scores among the classifiers for *imp_event_7days*.

Classifier	Accuracy	Recall	Precision	F1 Score
LogisticRegression	0.641538	0.624160	**0.738820**	0.676667
DecisionTreeClassifier	0.648245	0.705216	0.708221	0.706715
RandomForestClassifier	**0.683936**	0.779153	0.718616	0.747661
GradientBoostClassifier	0.667305	**0.922913**	0.659489	**0.769275**

The SHAP analysis provides valuable insight into feature impact on final output. SHAP results could change a lot based on whether sampling was done, indicating no strong trend in feature impact. The balanced distribution of red and blue dots across all features, regardless of their values, indicates no clear trend of a single feature dominating the predictions in any certain way. This equal spread highlights the challenge of identifying strong predictive patterns using these features.



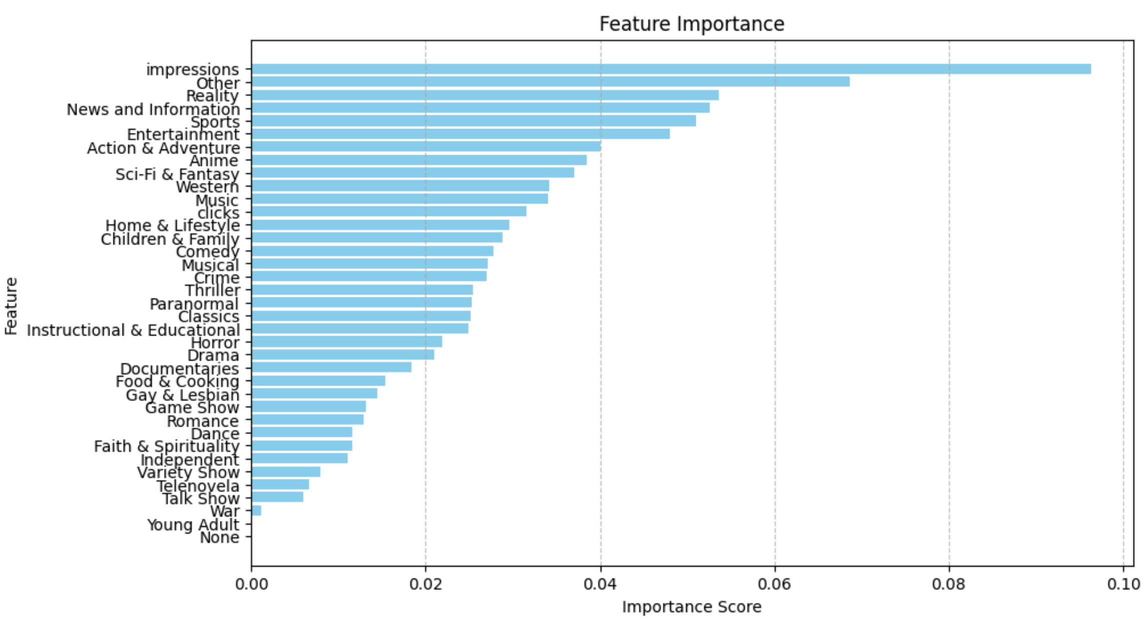


Average Minutes per User

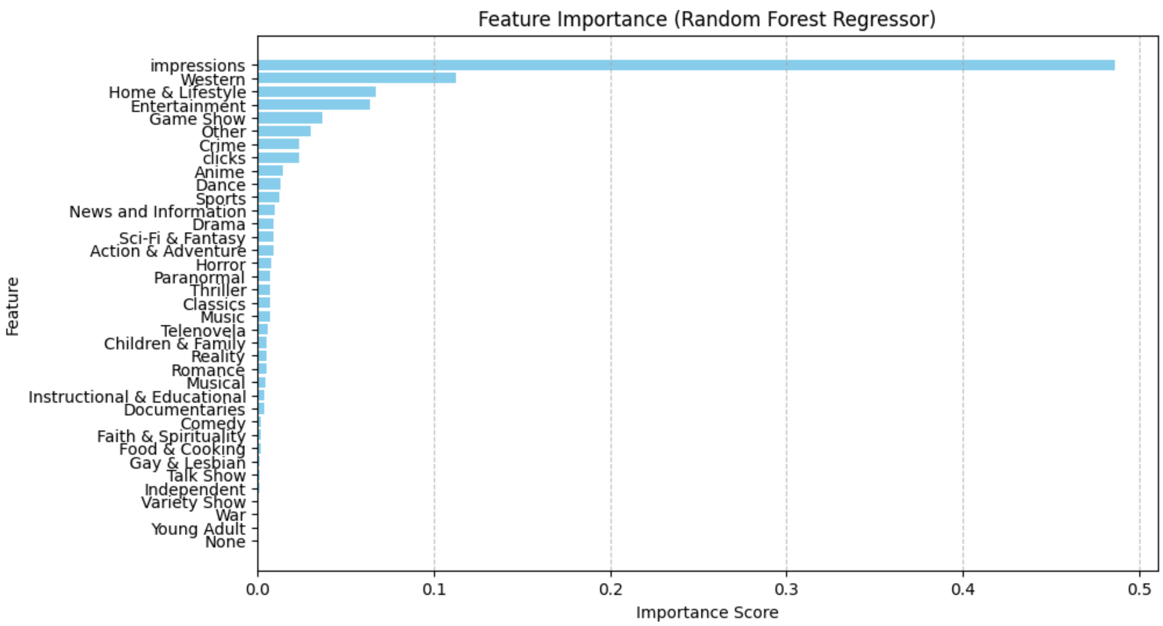
In order to determine success for predicting average minutes per user, the performance of two models were compared: Decision Tree Classifier and Decision Tree Regressor. For classification, Decision Tree Classifier was used to determine precision, recall, accuracy, and f1 score. For regression, Decision Tree Regressor success metrics used are mean squared error and r-squared.

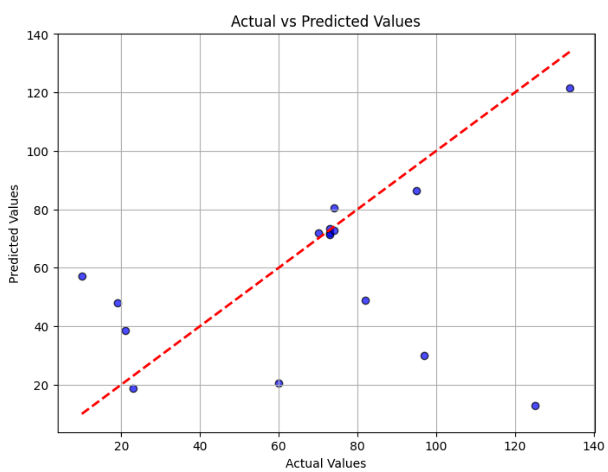
```
Accuracy: 0.56
Classification Report:
              precision    recall  f1-score   support

        Poor       0.25      0.33      0.29         3
        Fair       0.33      0.50      0.40         2
        Good       1.00      0.83      0.91         6
   Excellent       0.50      0.40      0.44         5

    accuracy                           0.56        16
   macro avg       0.52      0.52      0.51        16
weighted avg       0.62      0.56      0.58        16
```

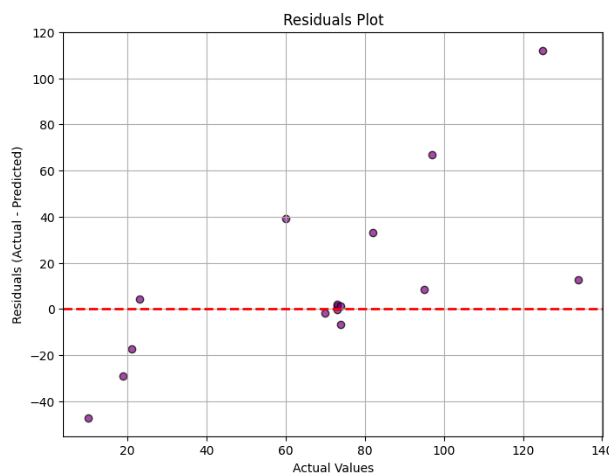
Decision Tree Classifier: The model achieved 56% accuracy, performing better than other models we tested. Precision varied: 25% for Poor, 33% for Fair, 100% for Good, and 50% for Excellent, indicating that the model excelled at predicting higher scores. Feature importance analysis confirmed that impressions had the largest impact, aligning with our expectations.



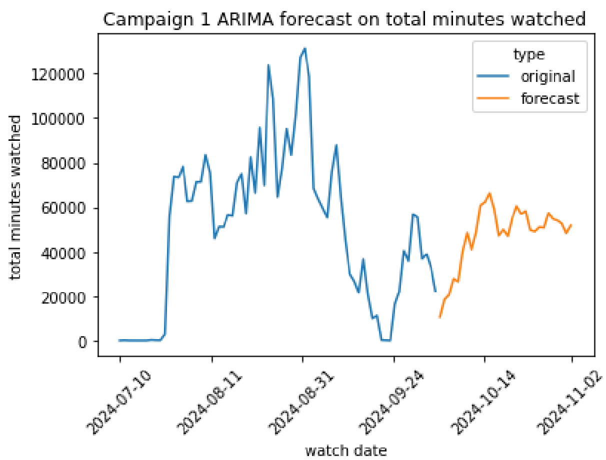
Decision Tree Regressor: Unfortunately, the model underperformed, with a mean absolute error of 24, a mean squared error of 1467, and an r-squared of -0.19. Attempts with linear models, neural networks, and other approaches showed similar results, indicating a lack of strong predictive features for this metric. However, as more historical data and features are added, future iterations may improve. Feature importance highlighted impressions as the dominant predictor, similar to what we observed with the decision tree classifier.

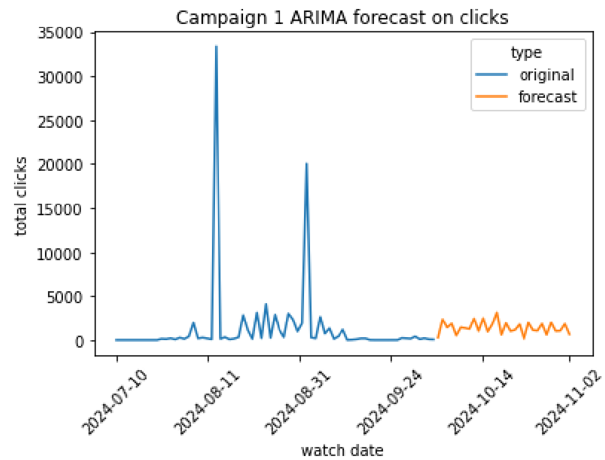






ARIMA predictive trend analysis: One of our goals was to create decision-making tools that could answer predictable questions about active campaigns. One such concept was to utilize ARIMA to make forecasts that could aid decision-making by presenting mathematical scenarios for a campaign's future performance based on its current trends. Due to general model accuracy issues, we focused on creating a function that (had we had more dashboard processing power) could be used to slide through lower to higher volatility scenarios, projecting key metrics such as clicks, impressions, and total minutes watched, a month into the future. On the current dashboard, this is the time-series data, but in future iterations, it might look something like this:



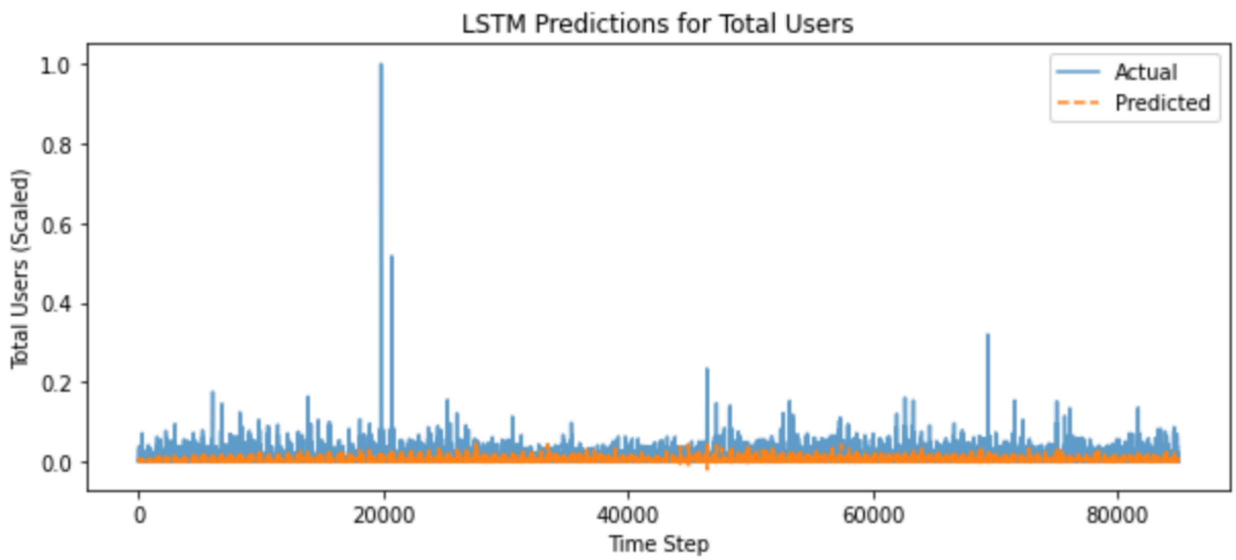


Deep Learning

Long Short Term Memory (LSTM): According to previous research by Sarkar and Bruyn, minimal feature engineering leads to more accurate results when comparing actual to predicted values. The only manipulation we used before feeding data into the model was to aggregate the dataset containing 8.6M records to total impressions, clicks, minutes watched, and users, resulting in a total of 85k records.

To evaluate the model, we used mean squared error (MSE), root mean squared error (RMSE), and mean absolute error (MAE). We can see that for all three metrics, there is error consistency, where smaller error values suggest a consistent and reliable model. This means that the LSTM has successfully captured the underlying patterns in the data, and its predictions can be trusted. One takeaway is that if MSE is significantly higher than MAE, suggesting that there are outliers or large errors in the predictions. We don't see that in our results.

MSE	RMSE	MAE
8.469712289925337e-05	0.009203103981769052	0.004193503932319161

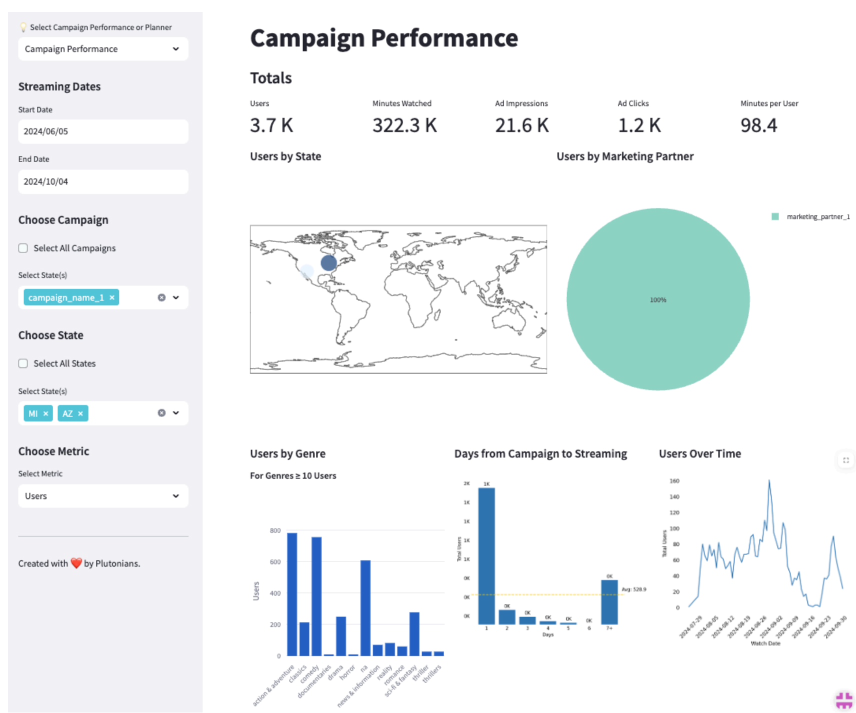


Interactive Solutions

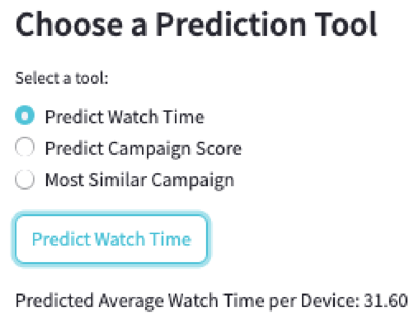
Dashboard Creation

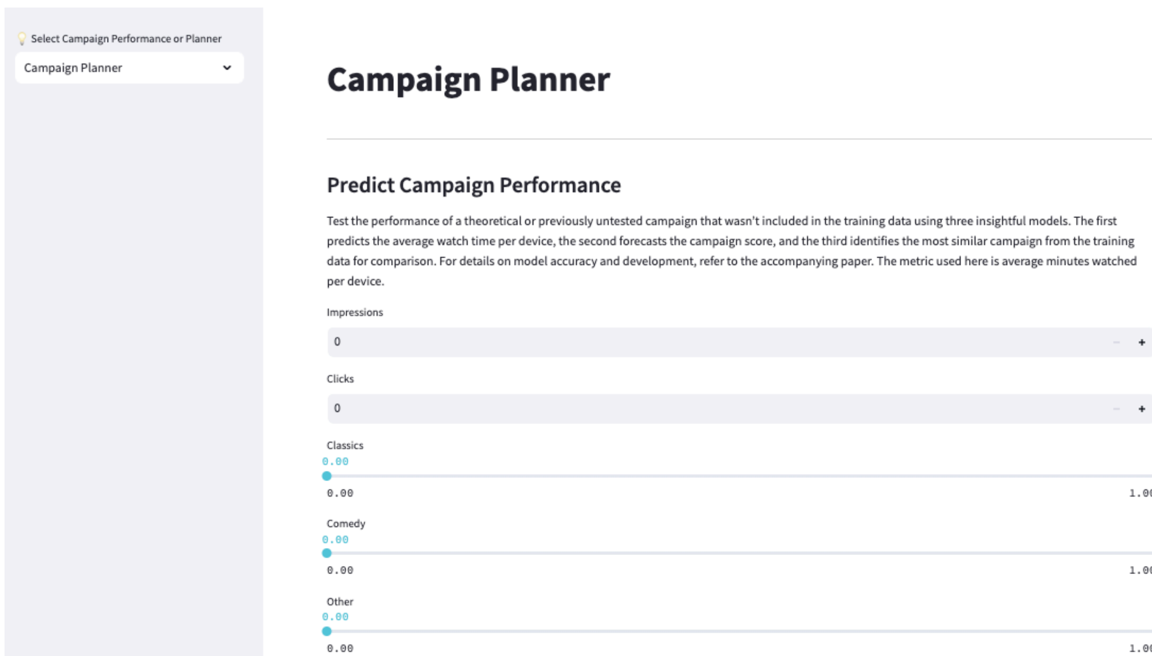
The modeling for this capstone project was designed with the ultimate goal of creating a user-friendly dashboard. The dashboard offers several key features:

Campaign Exploration: Users can analyze individual campaigns in detail, with breakdowns by state, genre, and key performance metrics, enabling insights into past campaigns.



ML-Powered Predictions: The dashboard leverages machine learning to predict campaign performance for new campaigns. Users can forecast metrics such as watch time and campaign scores, as well as identifying the most similar campaign. Once a similar campaign is identified, users can dive deeper into its performance using the campaign exploration features.





Discussion

Our team successfully developed a campaign analyzer and planner, integrating multiple machine learning models to enable predictive insights for optimizing future campaigns. The tool provides a comprehensive breakdown of key metrics, offering marketers a deeper understanding of campaign performance. Originally, the data comprised 8.6 million unclean rows, requiring extensive processing to create machine learning-ready datasets. We designed and implemented a robust ETL pipeline that transforms raw data into actionable insights, enabling state-level performance analysis with various filters and predictive capabilities.

Beyond the product itself, the project fostered valuable discussions about the process. While feature importance posed challenges, as no single feature emerged as highly successfully predictive, we successfully employed supervised, unsupervised, and deep learning models to extract meaningful insights. Our scalable design allows for future enhancements and seamless integration of additional data, ensuring adaptability to Pluto TV's evolving needs.

Comprehensive analysis and exploratory work, documented in our development repository, informed the final deliverable. While not all findings are directly reflected in the product, they laid the groundwork for creating a robust system. This deliverable not only includes the dashboard but also a powerful ETL process, preparing data for machine learning tasks and supporting future predictive testing.

Broader Impacts

As demonstrated above, we created a dynamic dashboard tool that allows PlutoTV marketers and management to better understand, evaluate, and predict advertising campaign success. Given that this tool will hopefully be directly used and implemented by PlutoTV, a free provider of streaming services with a large catalog of TV and film, we believe that creating a demonstrably useful tool to evaluate campaign effectiveness will largely have a positive impact. We don't believe that there are many risks to aiding the effort to provide free streaming services to more people, especially when TV used to be a widely provided free service. In a day and age where bundles and packages of streaming services have replaced what used to be a free service for anyone to use, regardless of economic level, we believe that PlutoTV is a positive outlier in the internet age.

That being said, a potential ethical concern is that the current users and campaign advertising targets may be affected by how their user data is being used to implement changes in advertising strategy. To prevent the leakage of user data, we anonymized it; however, there are risks that current demographic groups hidden within the data will shift where future funding goes, potentially away from users that could be positively impacted by free services. On the other hand, future advertising on the PlutoTV platform (not by the PlutoTV platform) could target the specific demographics that may be more likely to sign up for PlutoTV after seeing enhanced and more directed advertising campaigns. Another risk for Pluto TV marketers is understanding the predictive accuracy and reliability of the models. Since the models vary in success, users may unintentionally rely on less accurate predictions, leading to incorrect conclusions. Therefore, it is crucial for users to be aware of each model's performance metrics to make informed decisions and avoid misinterpreting campaign analysis results.

Closing Thoughts

We are pleased with the outcome of this project, which was a collaborative effort facilitated by regular team meetings, mentorship, and ongoing discussions via Slack. In conclusion, we successfully developed a Streamlit application for Pluto TV, enabling marketers to use new campaign data to predict and analyze campaign performance.

The application is scalable and extensible, allowing for easy integration of additional features by updating the ETL process with new datasets. Future extensions could include expanding data coverage, implementing deeper networks, adding more visualizations, or introducing a financial optimization module. Moreover, integrating a cloud database would improve performance by enabling faster data processing and storage, eliminating the need to work with local CSV files. In the long term, incorporating a chatbot to analyze campaigns could further enhance the application.

Despite time constraints, we are proud of our final product, which includes multiple machine learning models and a reusable, well-documented codebase. This foundation ensures the project is not only impactful but also ready for future iterations and advancements.

Statement of Work

Teammate	Contributions
Calvin Raab	• Set up the database using SQLite for fast data processing • Built the entire ETL pipeline to process 8.4 million rows, which created outputs which became the foundation for the team's workflow. • Aggregated data to the device level and further to the campaign level. • Prepared modeling-ready datasets, properly numbered and shared with the team. • Developed predictive models, including Nearest Neighbors, Random Forest Classifier, and Random Forest Regressor models, along with additional models tested but not ultimately used. • Conducted extensive feature engineering, exploratory data analysis (EDA), and model preparation. • Built the Streamlit application to integrate predictive models and designed the interface for campaign-level predictions. • Wrote comprehensive documentation on the ETL process, models, and predictive pipeline. • Created and managed the development GitHub repository for data and code sharing. • Built the dev GitHub repository and ensured smooth collaboration.
Chris Struck	• Implemented Kmeans and associated pair plots (visuals and analysis) as extended EDA of initial and final team dataframes • Assisted Calvin in updating and reviewing the processed dataset by creating dictionaries and evaluating state and genre results • Took lead in writing report; assigning sections and helping structure and edit the draft • Created ARIMA generation function for generation of future trend analysis of key features, including total minutes watched, impressions, and clicks • Created time series generation function utilized on the dashboard • Final grammar check and paper review
Justin Barry	• SHAP work for working with Naz file and Calvin file for top and

	bottom features of SHAP ● Some machine learning work ● Editing report and writing part of section about SHAP results ● Work on evaluation metrics of models ● Visuals for SHAP
Naz Uremek	● Led meetings with team and instructor ● Ensured the project timelines were met ● Developed code to gather initial raw data and combined into single csv file (8.4M records) ● Assisted Calvin with data encryption on the raw dataset to preserve sensitive information ● Combined team's visual inputs and models into an interactive tool using Streamlit Cloud ● Built the second Github repository for Streamlit only ● Wrote step by step readme guide in how to run the Streamlit app on their local computers ● Collaborated with the team on finalizing report ● Compared performance of classification models using supervised (LogisticRegression, DecisionTreeClassifier, RandomForestClassifier, GradientBoostClassifier), unsupervised (k-means), and deep learning (LSTM) methods

References

Koob, Clemens. “Determinants of Content Marketing Effectiveness: Conceptual Framework and empirical findings from a managerial perspective.” *PLOS ONE*, vol. 16, no. 4, 1 Apr. 2021, https://doi.org/10.1371/journal.pone.0249457.

Lopes, Ana, and Beatriz Casais. “Digital Content Marketing: Conceptual Review and Recommendations for Practitioners.” *Academy of Strategic Management Journal,* vol. 21, Jan. 2022, pp. 1-17, https://www.researchgate.net/publication/357746605_Digital_Content_Marketing_Conceptual_Review_and_Recommendations_for_Practitioners.

O’Sullivan, Don, and Andrew V Abela. “Marketing performance measurement ability and firm performance.” *Journal of Marketing*, vol. 71, no. 2, Apr. 2007, pp. 79–93, https://doi.org/10.1509/jmkg.71.2.79.

Sarkar, Mainak, and Arnaud De Bruyn. “LSTM response models for direct marketing analytics: Replacing feature engineering with Deep Learning.” *Journal of Interactive Marketing*, 2020, https://doi.org/10.2139/ssrn.3601025.

Appendix

Data Dictionary

Date Range: 6/5/24 - 10/4/24 (with data retention of 120 days at the time of data pull)

Region: US only

File format: csv

File size: ~2.5GB

Records: 8,671,816

Field	Datatype	Description	Example
device_id	string	Unique device ID, hashed (SHA256)	349e735c6c77b4d106f11ac98daddbaadd5f730a1170729ffa6ac89425a6f4f6
device_name	string	The app or device user streams Pluto TV on	iOS, Roku
device_type	string	Mobile = mobile device CTV = connected TV device	Mobile, CTV
impression_date	date	Date of ad impression	6/20/24
click_date	date	Date of ad click	6/26/24
install_date	date	Date when user installed Pluto TV	7/2/24
event_date	date	Date when user became an active user (at least 15 seconds of watched time)	7/10/24
marketing_channel	string	Marketing channel (i.e. paid social, paid search, programmatic, etc)	Apple Search Ads
marketing_partner	string	Marketing partner (i.e. Facebook, TikTok, Google, Reddit, etc)	Fire TV, Roku
paid_vs_organic	string	Whether or not a paid campaign was attributed to the install	paid, organic
campaign_id	string	Campaign ID of marketing campaign user is exposed to	123456789
campaign_name	string	Campaign name	Pluto TV - Summer of Cinema
adgroup_id	string	Adgroup ID of marketing campaign user is exposed to	234567890
adgroup_name	string	Ad group name	ASA Acquisition
creative_id	string	Creative ID of marketing campaign user is exposed to	345678901
creative_name	string	Creative name of ad	CS_PlutoTV_SOC_CTV_LG_Banners_June2024_4UP_MissionImpossible_970x250_v1.jpg
country	string	Two letter country code	US
state	string	US state (2 letter code)	MI
dma		Designated market area code	123
content_type	string	Live TV or view on demand (VOD)	live tv, vod
content_genre	string	Genre of content watched	Comedy, actions, sci-fi, drama, crime
channel_id	string	Channel identifier or 'vod' for view on demand	5b64a245a202b3337f09e51d
content_title	string	Title of content watched	sample_content_title
active_user	boolean	0=active user (watched content for 15 seconds or more during a single session*) 1=inactive user	0

		*Pluto TV's definition of session: • A new session starts when the user switches to another app, then back to Pluto TV • A new session starts when a user switches channels	
total_min_watched	integer	Total viewing minutes	60
impressions	integer	Total campaign ad impressions	2000
clicks	integer	Total campaign ad clicks	400
impressions_cost	integer	Total campaign spend from impressions	$100
clicks_cost	integer	Total campaign ad spend from clicks	$20