Ad Optimization: Improving CTR

Yu He, Haibu Shi,
Ellie Yang, Yuki Yu

Motivation

Problem Statement: Inefficiencies in digital ad targeting lead to wasted spending and low engagement

Solution: Optimization driven by Machine Learning with Multi-Armed Bandits

Business Impact: Improve CTR & Campaign efficiency

Data Overview

Kaggle Ad Click Prediction Dataset

- 4000 unique observations

Key Features:

- User attributes (gender, device, browsing behavior)
- Ad details (position, time of day)
- Click behavior (binary outcome)

1
Data Preprocessing

Preprocessing the data and creating customer segmentation

2
Machine Learning Modeling

Try out different ML models to predict ad click probability & best ad-placement

3
MAB

Dynamic ad-placement decisions

4
Future Directions

Next steps!

Data Preprocessing

01 Handling Duplicates & Missing Values

- Originally 10,000 rows but many duplicates → 4,000 unique observations
- Missing values: filled in with most frequent value based on the condition of another column

02 Feature Engineering

- Age grouping (binning)

03 Loss Reweighting

- **Logistic Regression:** class_weight='balanced' to adjust for class imbalance
- **XGBoost:** scale_pos_weight for higher penalties on underrepresented classes

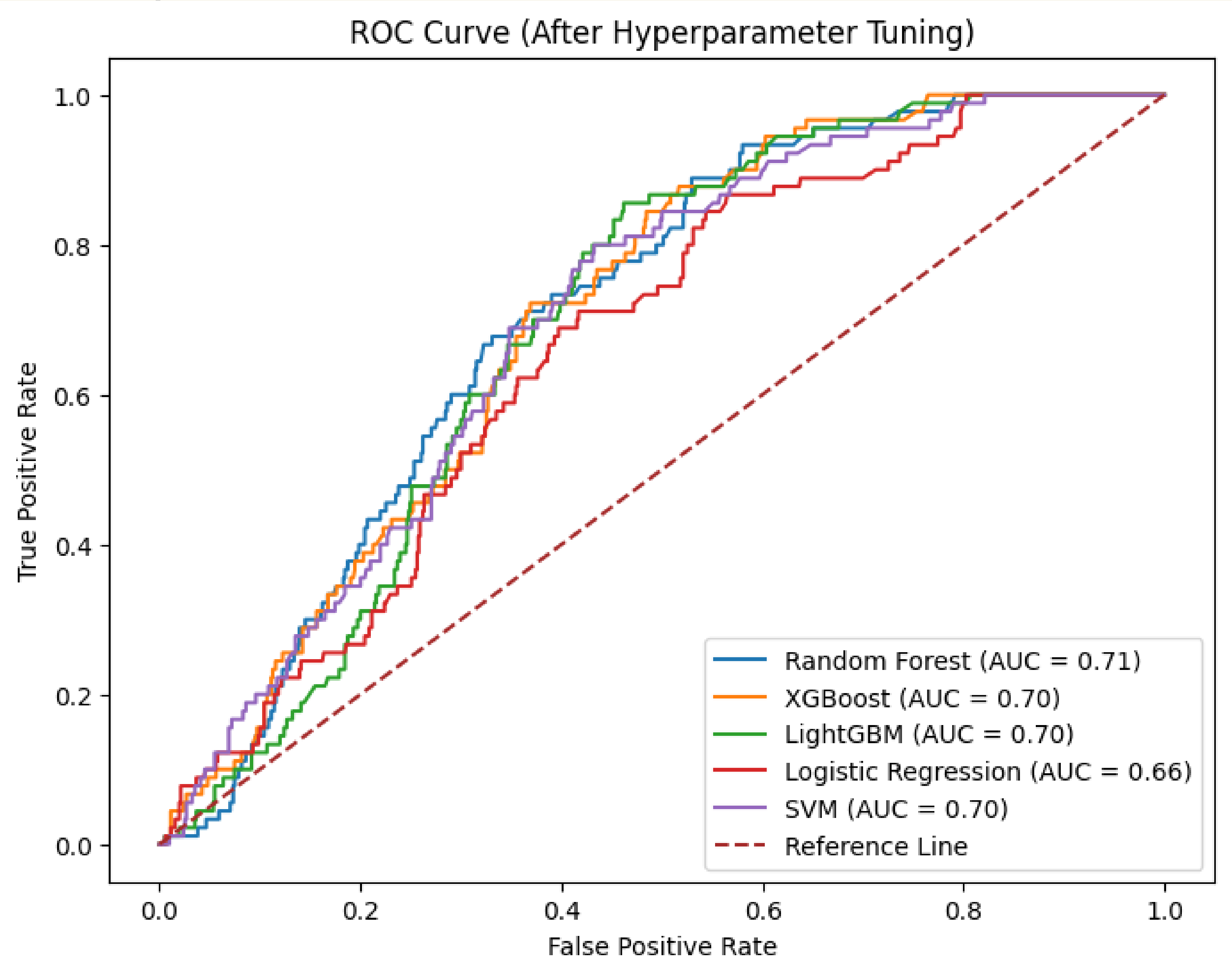
To better target customers:

Predict ad click probability

User attributes: gender, age, device type, browsing behavior
Ad details: position, time of day

1. **Logistic Regression**
2. **Support Vector Machine**
3. **Random Forest**
4. **XGBoost**
5. **LightGBM**

- **XGBoost & LightGBM** performed best, capturing nonlinear user behavior
- **Random Forest** had lower scores
- **SVM & Logistic Regression** less effective in capturing complex feature interactions



To determine ad placement:
Logistic Regression vs. XGBoost

Logistic Regression (Baseline)

- Assumes linear relationships
- Good baseline & interpretable
- Good at predicting top & Bottom ads
- Fast Computation

XGBoost

- Captures non-linear relationships & complex interactions
- Higher precision & recall
- Adapts to all ad placements
- Higher computational Cost

	Top		Bottom		Side	
Precision	0.517	0.537	0.559	0.585	0.364	0.418
Recall	0.609	0.691	0.681	0.702	0.122	0.176
Increase	+13.4%		+3.1%		+44.26%	

XGBoost is better for dynamic ad placement, while Logistic Regression is a strong, interpretable baseline.

With MAB, CTR is even better!

MAB

Why MAB?

- Traditional models fail to adapt to changing user engagement.
- MAB dynamically optimizes ad placement based on real-time feedback

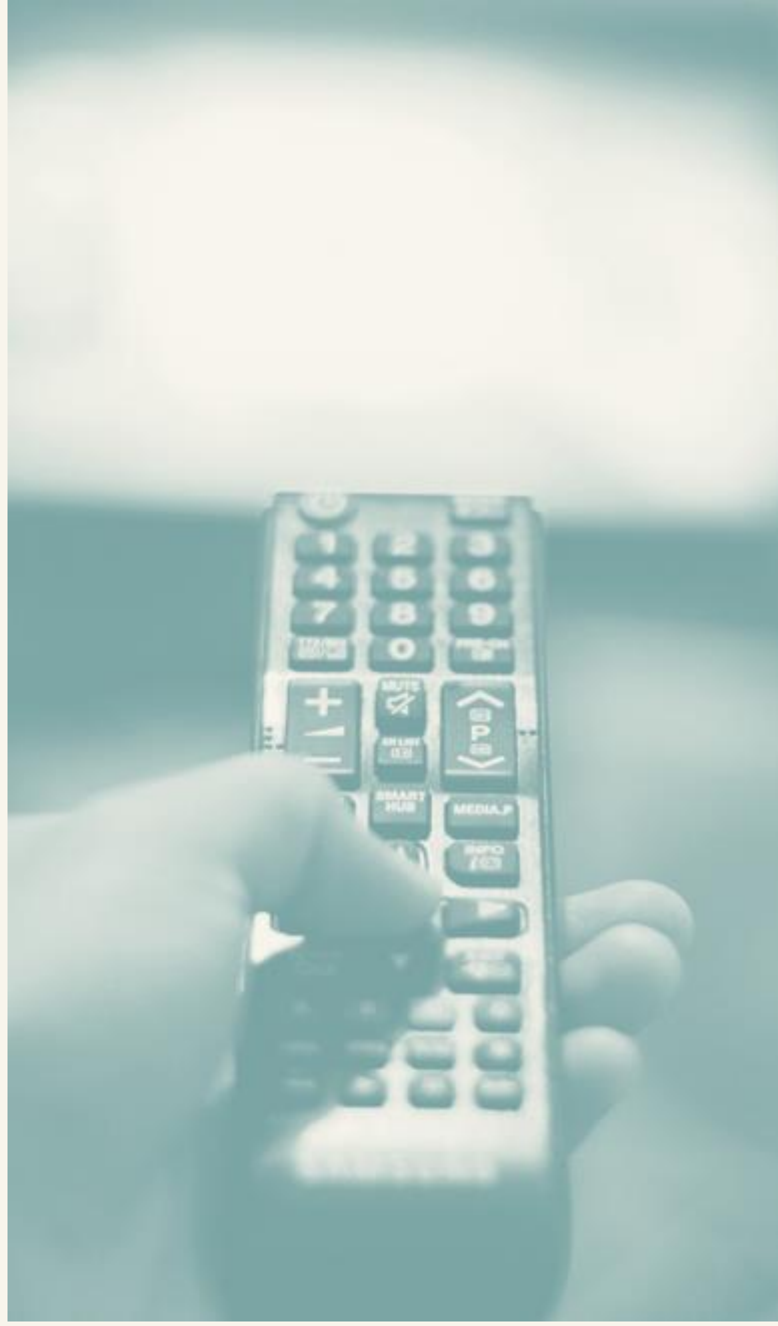
Key Factors Modeled

- **Seasonality:** CTR varies daily/weekly (modeled with trigonometric function)
- **Competitor Influence:** Ad effectiveness shifts due to market competition
- **Random Fluctuations:** Noise added to simulate real-world uncertainty

Epsilon-Greedy Strategy (ε = 0.05)

- **95%** → Chooses **highest CTR ad** (exploitation)
- **5%** → Selects **random ad** (exploration)

Prevents suboptimal decisions & adapts to new trends



CTR

Baseline Majority: 11.26% (Always ‘Top’)

MAB: 13.76% (More allocation to ‘Side’)

MAB outperformed the baseline, adapting in real-time to maximize CTR and ad efficiency.

Future Directions

A Hybrid Approach:

1. Use ML model to predict CTR for each ad placement 📊
2. Feed predictions into MAB for decision-making
   a. purely random exploration ❌
   b. prior knowledge as a starting point ✅
3. MAB learns & adapts!
   a. continuous improved decisions based on real-world feedback 🔄

Thank You!

Questions?

Comments?

Concerns?