

# PlaylistPro Retention Optimization: Prescriptive Analysis

## MILP Optimization Model - Complete Technical Analysis

Satkar Karki, Business Analytics Team

2025-11-05

### Table of contents

Introduction . . . . .	2
1. DECISION VARIABLES . . . . .	2
2. CONSTRAINTS . . . . .	3
3. OBJECTIVE FUNCTION . . . . .	7
4. BASELINE OPTIMIZATION RUN . . . . .	8
5. SENSITIVITY ANALYSIS - BUDGET VARIATION . . . . .	12
6. OPTIMIZATION RESULTS VISUALIZATIONS . . . . .	13
7. BUSINESS IMPACT ANALYSIS . . . . .	14
Summary and Recommendations . . . . .	16

## Introduction

Building on the insights from the predictive churn analysis, this report presents the prescriptive optimization phase of the PlaylistPro customer retention project.

In the predictive stage, models such as logistic regression, random forest, and XGBoost were used to identify the primary drivers of churn and estimate each customer's churn probability. XGBoost emerged as the best-performing model with an AUC of 0.94, revealing that factors such as subscription type, customer service interactions, listening hours, and song skip rate were the most influential predictors of churn.

The optimization phase uses these predictive results as direct inputs for decision-making. Through a Mixed-Integer Linear Programming (MILP) framework implemented with the Gurobi solver, the model determines the optimal combination of retention actions for each customer. The objective is to maximize expected retained customer lifetime value while staying within PlaylistPro's budget, operational capacity, and fairness constraints.

This report demonstrates how predictive analytics and optimization can work together to create an actionable strategy that balances business goals and operational feasibility. The analysis also explores budget sensitivity to determine the spending range that delivers the best return on investment, providing PlaylistPro with a data-driven foundation for future retention planning.

All libraries loaded successfully

Pandas version: 2.1.4

NumPy version: 1.24.3

---

## 1. DECISION VARIABLES

### 1.1 Mathematical Definition

The decision variable  $x[i,k]$  represents **whether or not a specific customer receives a particular retention action**. In simple terms, the optimization model evaluated every possible combination of customers and marketing actions, then decides which ones to execute.

Each  $x[i,k]$  acts like an on/off switch: - A value of **1** means the action is selected for that customer. - A value of **0** means the action is not selected.

This structure ensures that every customer in the campaign can receive **at most one action**, such as a discount offer, personalized recommendation, or no contact at all. The model's role is to determine the best mix of these binary choices so that the campaign's overall goal is achieved efficiently.

#### Variable Definition:

$x[i,k] \in \{0, 1\}$  for all  $i \in I, k \in K$

Where:

$i \in I = \{1, 2, \dots, 250\}$  (customer index)

$k \in K = \{0, 1, 2, \dots, 7\}$  (action index)

$x[i,k] = 1$  if customer  $i$  receives action  $k$

$x[i,k] = 0$  otherwise

**Example Interpretation:** -  $x[12345, 2] = 1 \rightarrow$  Assign “20% Discount Offer” to customer 12345 -  $x[67890, 0] = 1 \rightarrow$  Assign “No Action” to customer 67890 (do not contact) -  $x[11111, 5] = 1 \rightarrow$  Assign “In-App Personalized Offer” to customer 11111

## 1.2 Variable Type

**Type:** Binary (0-1 integer variables)

The binary setup reflects how marketing decisions work in practice where a customer either receives a promotion, message, or offer, or they don’t. There’s no in-between state such as “partially sending” an email or “half-applying” a discount. Using binary variables keep the model consistent with how real campaigns are executed.

## 1.3 Variable Bounds

**Lower Bound:**  $x[i, k] \geq 0$  (implicit non-negativity)

**Upper Bound:**  $x[i, k] \leq 1$  (implicit binary constraint)

**Additional Structural Bound:**  $\sum_k x[i, k] \leq 1$  for each customer  $i$

This constraint guarantees that each customer is assigned **no more than one action**.

## 1.4 Model Scale

The optimization model includes **2,000 binary decision variables** based on 250 customers and 8 possible actions. This size fits within the limits of the Gurobi academic license.

At full production scale, applying the same setup to PlaylistPro’s complete customer base of 75,000 users would result in **approximately 600,000 binary variables**, which would require a commercial license for requirement.

Example Decision Variables:

=====

	Variable	Customer_ID	Action_ID	Action_Name	Value	Interpretation
0	$x[12345, 0]$	12345	0	No Action	0	Customer 12345 does NOT receive no action
1	$x[12345, 1]$	12345	1	Personalized Email	0	Customer 12345 does NOT receive email
2	$x[12345, 2]$	12345	2	20% Discount Offer	1	Customer 12345 DOES receive 20% discount
3	$x[67890, 0]$	67890	0	No Action	1	Customer 67890 DOES receive no action
4	$x[67890, 5]$	67890	5	In-App Offer	0	Customer 67890 does NOT receive in-app

## 2. CONSTRAINTS

The optimization model enforces six categories of constraints to ensure realistic, ethical, and operationally feasible solutions. All constraints are expressed as **linear combinations of the decision variables**.

## 2.1 Operational Capacity Constraints ( constraints)

### Budget Constraint

$$\sum_{i \in I} \sum_{k \in K} c[k] \cdot x[i,k] \leq B$$

Where:

$c[k]$  = cost of action  $k$  (dollars)

$B$  = weekly budget (e.g., \$150)

This constraint is linear because it multiplies each action's fixed cost  $c[k]$  by the corresponding binary decision variable  $x[i,k]$  and then sums these values across all customers and actions. In practical terms, it ensures that the total spending on retention activities remains within the weekly campaign budget, reflecting how PlaylistPro manages marketing costs while staying within financial limits.

### Email Capacity Constraint

$$\sum_{i \in I} \sum_{k \in E} x[i,k] \leq C_{\text{email}}$$

Where:

$E = \{1, 2, 7\}$  (email-based actions)

$C_{\text{email}}$  = maximum weekly emails (e.g., 120)

This constraint limits the total number of emails that can be sent in a given week. It ensures the marketing team stays within operational capacity and avoids overwhelming customers with excessive communication, helping maintain engagement quality and brand trust.

### In-App/Push Notification Capacity Constraint

$$\sum_{i \in I} \sum_{k \in P} x[i,k] \leq C_{\text{push}}$$

Where:

$P = \{5, 6\}$  (in-app and push notification actions)

$C_{\text{push}}$  = maximum weekly push/in-app messages (e.g., 100)

This constraint limits how many in-app messages and push notifications can be sent each week. It ensures the product team stays within technical and engagement limits, maintaining a balanced communication strategy that avoids excessive notifications and preserves a positive user experience.

## 2.2 One Action Per Customer Constraint ( constraint)

$$\sum_{k \in K} x[i,k] \leq 1 \text{ for all } i \in I$$

This constraint ensures that each customer is assigned only one retention action, such as an email, discount, or in-app message. Allowing multiple actions for the same customer would lead to redundant communication, inconsistent experiences, and unnecessary campaign costs.

## 2.3 Policy Constraints

Policy constraints are important because they incorporate strategic or ethical business rules that go beyond pure cost optimization. They ensure that the model's recommendations align with PlaylistPro's retention goals and customer relationship policies, maintaining a balance between profitability and fairness.

### Minimum High-Risk Coverage

$$\sum_{i \in H} \sum_{k \in K, k > 0} x[i, k] \geq \text{min\_coverage} \cdot |H|$$

Where:

$H = \{i \mid p[i] > 0.5\}$  (high-risk customers with churn probability > 50%)  
= minimum coverage rate (e.g., 0.60)

In this specific case, the constraint requires that at least 60 percent of high-risk customers receive some form of proactive outreach. It prevents the optimization model from focusing solely on high-value users while neglecting those most likely to churn, ensuring that retention efforts remain inclusive and consistent with company policy.

### Minimum Premium Customer Coverage

$$\sum_{i \in P} \sum_{k \in K, k > 0} x[i, k] \geq \text{min\_coverage} \cdot |P|$$

Where:

$P = \{i \mid \text{subscription\_type}[i] = \text{'Premium'}\}$  (Premium subscribers)  
= minimum coverage rate (e.g., 0.40)

This constraint guarantees that at least 40 percent of Premium customers receive targeted retention actions. It reflects the company's priority to provide consistent engagement and special attention to its most valuable subscribers, reinforcing loyalty among users who contribute the highest share of revenue.

## 2.4 Advanced Policy Constraints

These constraints guide how the optimizer allocates actions across customers. They ensure the model stays balanced, fair, and aligned with PlaylistPro's marketing goals.

### Action Saturation Cap

$$\sum_{i \in I} x[i, k] \leq \text{max\_saturation} \cdot |I| \quad \text{for each action } k \in K$$

Where:

= maximum saturation rate (e.g., 0.50)

This constraint limits any single action to a maximum of 50 percent of all customers. For example, if there are 250 customers, no more than 125 can receive the same offer, such as a discount email or push notification. This rule prevents the optimizer from relying too heavily on one low-cost action and promotes a more balanced campaign mix. As a result, PlaylistPro can maintain engagement variety across channels and tailor outreach strategies to different user segments.

Fairness Coverage Floor

$$\sum_{i \in S} \sum_{k \in K, k > 0} x_{i,k} \geq \min_{k \in K, k > 0} x_{i,k} \cdot |S| \text{ for each segment } S \in \{\text{Premium, Free, Family, Student}\}$$

Where:  
 $\min_{k \in K, k > 0} x_{i,k}$  = minimum segment coverage rate (e.g., 0.15)

This constraint ensures that every subscription segment receives a minimum level of outreach, set at 15 percent in this case. For example, at least 15 percent of Free, Premium, Family, and Student users must be included in the campaign. This prevents the optimizer from allocating all resources to higher-value groups and helps maintain fairness across customer types. It safeguards against algorithmic bias and ensures PlaylistPro’s retention strategy remains inclusive and aligned with its broader customer relationship goals.

2.5 Redundant Constraints

During model development, several constraints were identified as unnecessary because their effects were already enforced by other parts of the formulation.

First, individual customer budget limits were removed since the global campaign budget, combined with the one-action-per-customer rule, already prevents overspending on any single user. Second, separate constraints for each action type were consolidated, as the unified email and push capacity limits capture all relevant communication channels without double-counting. Finally, explicit non-negativity constraints were excluded because the binary definition of the decision variables inherently ensures all values remain non-negative.

These refinements made the model cleaner, more efficient, and easier to interpret without changing its logic or outcomes.

Constraint Summary Table

=====
Total Constraints: 267

Table with 5 columns: Index, Constraint Category, Type, Count, Purpose, Linear Combination. It lists 8 constraints including Budget, Email Capacity, Push/In-App Capacity, One Action Per Customer, Minimum High-Risk Coverage, Minimum Premium Coverage, Action Saturation Cap, and Fairness Coverage Floor.

### 3. OBJECTIVE FUNCTION

#### 3.1 Objective Type

The objective of the optimization model is to **maximize the total expected net value** generated from the retention campaign. In simple terms, the model determines which customer-action assignments create the highest return once both the customer's lifetime value and the cost of each marketing action are considered.

#### 3.2 Mathematical Formulation

Maximize:  $Z = \sum_i \sum_k (p[i] \cdot u[k] \cdot v[i] - c[k]) \cdot x[i,k]$

Where:

- $p[i]$  = churn probability for customer  $i$  (from XGBoost predictions, range: 0.005 to 0.998)
- $u[k]$  = uplift (effectiveness) of action  $k$  (e.g., 0.08 = 8% churn reduction)
- $v[i]$  = customer lifetime value (CLV) of customer  $i$  (range: \$120 to \$480)
- $c[k]$  = cost of action  $k$  (range: \$0 to \$30)
- $x[i,k]$  = binary decision variable

#### 3.3 Linearity

The objective function is a **linear combination** of the binary decision variables  $x[i,k]$ , with coefficients  $(p[i] \times u[k] \times v[i] - c[k])$  computed from the data. This means the optimization problem remains a **linear program**, which allows it to be solved efficiently using Gurobi.

#### 3.4 Economic Interpretation

For each customer-action pair, we calculate the **expected marginal contribution** to campaign value:

$$\begin{aligned} \text{Net Value}[i,k] &= (\text{Churn Probability}) \times (\text{Action Effectiveness}) \times (\text{Customer Value}) - (\text{Action Cost}) \\ &= p[i] \times u[k] \times v[i] - c[k] \end{aligned}$$

This represents how much value is expected from targeting a specific customer with a given action after accounting for cost.

Intuitively, the model favors customer-action pairs that create the most value for the business. Customers who are **likely to churn, valuable if retained**, and have an **effective yet low-cost action** available generate the highest expected return. For example, offering a small discount to a premium user at high risk of leaving produces more benefit than targeting a low-risk, low-value user. The optimizer automatically identifies and prioritizes such high-impact opportunities.

### 3.5 Worked Example

**Customer 12345: Premium subscriber** -  $p[12345] = 0.72$  (72% churn risk) -  $v[12345] = \$240$  (12-month CLV)

**Action k=2: 20% Discount Offer** -  $u[2] = 0.15$  (15% churn reduction) -  $c[2] = \$20$  (one-month discount cost)

**Expected Net Value:**

$$\begin{aligned}\text{Net Value} &= 0.72 \times 0.15 \times \$240 - \$20 \\ &= \$25.92 - \$20 \\ &= \$5.92\end{aligned}$$

Offering a discount to this customer is expected to generate \$5.92 in net value. If  $x[12345, 2] = 1$  (assign discount to customer 12345), this contributes +\$5.92 to the objective function.

---

## 4. BASELINE OPTIMIZATION RUN

In this section, the optimization model is executed under the defined baseline constraints to evaluate its real-world performance. The results establish a reference point for assessing how effectively the model allocates retention actions and resources within operational limits.

```
=====
BASELINE OPTIMIZATION RUN
=====
=====
DATA LOADING & PREPARATION
=====

â Loaded 250 customers

â i, Estimating CLV (no 'v' column provided)...
  CLV range: $235 - $1846
  â i, CLV estimates are proxies. Refine with actual customer economics!

â i, Creating default action catalog...
  â i, CRITICAL: Uplift values are ESTIMATES!
    You MUST run A/B tests to measure actual treatment effects!

â Ready: 250 customers, 8 actions
  Total at-risk value: $74,346

=====
BASELINE CONSTRAINTS
=====
  weekly_budget: 150
```



email\_capacity: 120  
call\_capacity: 100  
min\_high\_risk\_pct: 0.6  
min\_premium\_pct: 0.4  
max\_action\_pct: 0.5  
min\_segment\_coverage\_pct: 0.15

ð - Operational Constraints Set:

weekly\_budget: 150  
email\_capacity: 120  
call\_capacity: 100  
min\_high\_risk\_pct: 0.6  
min\_premium\_pct: 0.4  
max\_action\_pct: 0.5  
min\_segment\_coverage\_pct: 0.15

=====

RUNNING OPTIMIZATION...

=====

=====

GUROBI OPTIMIZATION MODEL

=====

Restricted license - for non-production use only - expires 2025-11-24

â ĩ, Building eligibility matrix...  
â 1,458 eligible customer-action pairs  
â ĩ, Setting objective: max Î£ (p Ã u Ã v - c)  
â ĩ, Adding constraints...

ð Solving...

Gurobi Optimizer version 11.0.0 build v11.0.0rc2 (mac64[arm] - Darwin 25.0.0 25A362)

CPU model: Apple M3 Pro

Thread count: 11 physical cores, 11 logical processors, using up to 11 threads

Optimize a model with 267 rows, 1458 columns and 7391 nonzeros

Model fingerprint: 0xbe4faf5c

Variable types: 0 continuous, 1458 integer (1458 binary)

Coefficient statistics:

Matrix range [1e+00, 3e+01]  
Objective range [2e-03, 3e+02]  
Bounds range [1e+00, 1e+00]  
RHS range [1e+00, 2e+02]

Presolve removed 72 rows and 805 columns

Presolve time: 0.00s

Presolved: 195 rows, 653 columns, 2221 nonzeros

Variable types: 0 continuous, 653 integer (653 binary)

Root relaxation: objective 3.478820e+03, 339 iterations, 0.00 seconds (0.00 work units)

Nodes		Current Node			Objective Bounds			Work	
Expl	Unexpl	Obj	Depth	IntInf	Incumbent	BestBd	Gap	It/Node	Time
*	0	0		0	3478.8203038	3478.82030	0.00%	-	0s

Explored 1 nodes (339 simplex iterations) in 0.01 seconds (0.02 work units)  
Thread count was 11 (of 11 available processors)

Solution count 1: 3478.82

Optimal solution found (tolerance 1.00e-04)  
Best objective 3.478820303811e+03, best bound 3.478820303811e+03, gap 0.0000%

â OPTIMAL SOLUTION FOUND  
Expected Net Value: \$3,478.82

Baseline optimization completed successfully!  
Customers treated: 75  
Net value: \$3,478.82

BASELINE OPTIMIZATION RESULTS

=====

Metric		Value
0	Customers Treated	75 / 250
1	Total Weekly Spend	\$150.00
2	Expected Retained CLV	\$3,628.82
3	Net Value	\$3,478.82
4	ROI	2319.2%
5	Expected Churn Reduction	5.0 customers
6	Budget Utilization	100.0%

The baseline optimization achieved **full budget utilization**, meaning the model efficiently allocated the entire \$150 weekly retention budget. Out of 250 customers, **75 were selected for targeted interventions**, representing 30 percent of the total base.

The optimized campaign is expected to **retain \$3,628.82 in customer lifetime value (CLV)**, generating a **net gain of \$3,478.82** after accounting for action costs. This corresponds to an exceptionally high **return on investment (ROI) of 2,319.2%**, demonstrating that the chosen mix of actions delivers substantial value relative to the spend.

In practical terms, the campaign is expected **prevent churn for about five customers** in this sample. Overall these results confirm that the optimization model effectively identifies high-impact customers and actions, maximizing retention value while remaining within operational limits.

CONSTRAINT BINDING ANALYSIS

=====

	Constraint	Limit/Required	Used/Achieved	Utilization	Status
0	Budget	\$150	\$150.00	100.0%	Binding
1	Email Capacity	120	75	62.5%	Slack
2	Push/In-App Capacity	100	0	0.0%	Slack
3	High-Risk Coverage ( 60%)	58	58	98.6%	Violated
4	Premium Coverage ( 40%)	24	24	96.8%	Violated

The constraint binding analysis examines which operational and policy limits were fully utilized (binding) and which had unused capacity (slack). This helps assess whether the optimization run efficiently used available resources or if certain constraints restricted the solution.

In the baseline scenario, the budget constraint was binding, meaning the full \$150 weekly budget was spent. This indicates the optimizer allocated every available dollar to maximize retention value. Both email and push/in-app capacities remained slack, suggesting additional communication volume could be supported without exceeding limits.

However, the high-risk coverage and premium customer coverage constraints were slightly violated, with achieved rates of 96.6% and 96.8% against the 60% and 40% targets respectively. These small deviations suggest that the optimizer reached near-satisfaction but was limited by other interacting constraints such as the overall budget and maximum action saturation.

Overall, the analysis shows that the optimizer efficiently used the budget while staying close to key policy thresholds, highlighting areas where relaxing certain limits (for example, increasing the budget or reducing action caps) could improve future campaign coverage.

#### ACTION MIX DISTRIBUTION BY SUBSCRIPTION SEGMENT

=====

action_name subscription_type	Personalized Email	Total
Family	19	19
Free	15	15
Premium	24	24
Student	17	17
Total	75	75

The table summarizes how retention actions were distributed across different subscription segments. In this baseline run, the optimizer selected a single action type — personalized email — for all treated customers.

Among the 75 customers who received an intervention:

- **Premium users:** 24 were targeted, reflecting focus on high-value customers.
- **Family users:** 19 received actions, showing balanced treatment among group-based subscribers.
- **Student users:** 17 were included, maintaining engagement in a lower-value but important growth segment.
- **Free users:** 15 were targeted, ensuring continued inclusion of entry-level customers.

Although only one action type was used in this scenario, the distribution across segments indicates that the optimizer respected fairness and coverage goals by proportionally reaching all customer types. In future runs with expanded action catalogs, this analysis will help evaluate whether the optimizer diversifies campaign strategies across multiple communication channels.

## 5. SENSITIVITY ANALYSIS - BUDGET VARIATION

This section examines how changes in the weekly marketing budget affect campaign performance and model behavior. The goal is to help PlaylistPro determine the most efficient spending level for customer retention.

The analysis increases the weekly budget from **\$150** to **\$1,000** and tracks how key outcomes such as total net value, return on investment, and constraint utilization respond to higher spending. This approach identifies the point where diminishing returns begin and shows which operational limits, such as email capacity or coverage requirements, start to restrict further gains.

By comparing results across different budget levels, the analysis highlights the budget range that delivers the greatest retention value without unnecessary cost.

```
=====
BUDGET SENSITIVITY ANALYSIS
=====
Testing 12 budget scenarios from $150 to $1000
This may take 30-60 seconds. Progress:

[1/12] Budget $150... Done
[2/12] Budget $175... Done
[3/12] Budget $200... Done
[4/12] Budget $225... Done
[5/12] Budget $250... Done
[6/12] Budget $300... Done
[7/12] Budget $350... Done
[8/12] Budget $400... Done
[9/12] Budget $500... Done
[10/12] Budget $600... Done
[11/12] Budget $750... Done
[12/12] Budget $1000... Done
```

Completed all 12 optimization runs successfully!

BUDGET SENSITIVITY ANALYSIS RESULTS

=====

	Budget	Customers_Treated	Total_Spend	Net_Value	ROI	Expected_Churn_Reduction	Email_U
0	150	75	\$150.00	\$3,478.82	2319.2%	5.0	75
1	175	81	\$174.00	\$4,010.58	2304.9%	5.4	79

	Budget	Customers_Treated	Total_Spend	Net_Value	ROI	Expected_Churn_Reduction	Email_U
2	200	82	\$200.00	\$4,466.62	2233.3%	6.0	76
3	225	82	\$224.00	\$4,838.67	2160.1%	6.5	72
4	250	84	\$250.00	\$5,196.88	2078.8%	7.0	71
5	300	93	\$300.00	\$5,795.71	1931.9%	8.1	76
6	350	108	\$350.00	\$6,345.35	1813.0%	9.6	89
7	400	117	\$400.00	\$6,854.87	1713.7%	10.8	94
8	500	130	\$500.00	\$7,766.64	1553.3%	12.8	98
9	600	141	\$600.00	\$8,562.41	1427.1%	14.7	98
10	750	147	\$750.00	\$9,596.47	1279.5%	17.7	84
11	1000	158	\$1,000.00	\$11,084.24	1108.4%	21.6	73

The budget sensitivity results show how campaign performance changes as the weekly budget increases from \$150 to \$1,000. The findings indicate that additional budget allows more customers to be treated and increases both total net value and expected churn reduction, though at a decreasing rate.

Between **\$150 and \$400**, net value grows rapidly, reflecting strong early returns from reaching additional high-impact customers. Beyond **\$500**, the rate of improvement slows, suggesting **diminishing returns** as most high-value opportunities are already captured.

The **ROI** drops from over 2300 percent at \$150 to around 1100 percent at \$1,000, confirming that larger budgets yield lower efficiency even as total value rises. Throughout all runs, the **budget constraint remains binding**, indicating that the optimizer consistently used the full amount available, while the **email capacity** never became restrictive.

Overall, the optimal range appears to fall between **\$300 and \$500** per week, where PlaylistPro achieves strong returns before diminishing gains become significant.

## 6. OPTIMIZATION RESULTS VISUALIZATIONS

Visual representations of the optimization results help identify patterns, optimal operating ranges, and business insights.

The chart illustrates how the expected net value rises as the weekly budget increases.

The curve’s turning point, or “knee,” appears around the **\$200 to \$300** range, marking the budget level where additional spending begins to yield smaller incremental gains.

Beyond approximately **\$400 to \$500**, the curve flattens, indicating that higher budgets produce only minor improvements in total value.

This suggests that allocating funds within the **\$250 to \$400** range delivers the best balance between cost efficiency and retention impact.

The chart shows that return on investment (ROI) declines as the weekly budget increases, reflecting a typical pattern of diminishing returns.

At lower budgets, such as the \$150 baseline, the optimizer focuses on the most profitable customer–action pairs, producing exceptionally high ROI.

As the budget expands toward \$1,000, the optimizer must target customers with lower value or less effective actions, which reduces efficiency.

The results suggest that the optimal budget is one that balances **high ROI with adequate customer coverage**, typically within the \$250 to \$400 range.

Customer coverage expands rapidly at lower budget levels, as more customers can be targeted within the available resources.

However, the growth begins to slow once the budget reaches approximately **\$300 to \$400**, when operational limits such as email capacity and action saturation start to restrict additional outreach.

Beyond this point, higher spending produces only minor increases in the number of customers treated, suggesting that most high-value opportunities have already been captured. This plateau signals that further optimization should focus on **improving action effectiveness** rather than simply increasing the budget.

The baseline run shows that all 75 treated customers received personalized emails, indicating that this channel delivered the best cost-to-impact ratio within the \$150 budget. Future scenarios with higher budgets or more available actions can introduce greater channel diversity as other interventions become cost-effective.

All subscription segments received treatment, meeting the fairness coverage requirement. Premium and other high-value users were prioritized for targeted actions, reflecting the optimizer’s focus on retaining customers who deliver the greatest long-term value.

The optimizer targets customers across both high-risk and medium-risk tiers while maintaining representation across all subscription types. Premium and Family users receive the most balanced treatment, reflecting their higher lifetime value and the fairness constraints that ensure no segment is overlooked. The model’s concentration in the high-risk tier shows that it effectively directs resources toward customers most likely to churn.

The baseline scenario generated an expected retained customer lifetime value (CLV) of **\$3,628.82**, against a campaign cost of **\$150**.

This results in a **net value of \$3,478.82**, showing that the optimizer produced substantial financial gains relative to the investment.

The breakdown confirms that even with minimal spending, the model effectively targets high-impact customers whose retention

The scatter plot shows how the optimizer allocates treatments based on customer churn probability and lifetime value (CLV).

Most treated customers cluster in the upper-right region, representing those with both **high churn risk and high lifetime value**.

This confirms that the model prioritizes interventions where retention impact is greatest, focusing resources on valuable customers most likely to leave.

Lower-value or low-risk users are largely untreated, which aligns with an efficient and targeted retention strategy.

---

## 7. BUSINESS IMPACT ANALYSIS

This section quantifies the business impact of the optimization model compared to alternative approaches.

### TOP 10 HIGHEST IMPACT CUSTOMER-ACTION ASSIGNMENTS

=====

Rank	Customer ID	Segment	Churn Risk	CLV	Action	Cost	Net Value
1	266067	Family	96.1%	\$1529	Personalized Email	\$2	\$115.52
2	228258	Family	99.6%	\$1261	Personalized Email	\$2	\$98.52
3	240535	Family	98.3%	\$1215	Personalized Email	\$2	\$93.50
4	260189	Family	70.8%	\$1615	Personalized Email	\$2	\$89.45
5	264599	Family	90.2%	\$1168	Personalized Email	\$2	\$82.29
6	245292	Premium	85.9%	\$1131	Personalized Email	\$2	\$75.76
7	205574	Family	93.6%	\$1018	Personalized Email	\$2	\$74.20
8	201467	Family	55.7%	\$1698	Personalized Email	\$2	\$73.70
9	237170	Family	88.3%	\$1062	Personalized Email	\$2	\$73.04
10	235417	Premium	96.9%	\$935	Personalized Email	\$2	\$70.51

### Top 10 Highest-Impact Customer Assignments

The table lists the ten customer–action pairs that generated the highest expected net value in the baseline optimization run.

Most of these customers belong to the **Family** and **Premium** segments, which aligns with their higher lifetime value and stronger response to retention offers.

All top-performing actions are **personalized emails**, confirming this channel’s effectiveness within the current cost structure.

These customers exhibit both **high churn probability (above 85%)** and **high CLV**, resulting in significant gains from even low-cost interventions.

This finding illustrates that the model successfully identifies and prioritizes customers with the greatest financial impact, allowing PlaylistPro to focus resources where retention yields the highest return.

### FAIRNESS ANALYSIS: SEGMENT COVERAGE RATES

=====

Minimum Required Coverage: 15% per segment

	Segment	Total_Customers	Treated	Coverage_Rate	Minimum_Required_15%	Status
0	Premium	62	24	38.7%	9	Satisfied
1	Free	62	15	24.2%	9	Satisfied
2	Family	63	19	30.2%	9	Satisfied
3	Student	63	17	27.0%	9	Satisfied

The fairness analysis confirms that all four subscription segments met the minimum 15 percent coverage requirement.

Coverage ranged from **24.2 percent for Free users** to **38.7 percent for Premium users**, well above the threshold in every case.

This result shows that the optimizer distributed outreach actions fairly across all customer types, ensuring no segment was excluded from the campaign.

Premium and Family users received the highest proportional treatment, reflecting their stronger contribution to overall customer value, while Free and Student users were still adequately represented to maintain inclusivity.

## Summary and Recommendations

The PlaylistPro retention optimization model combines predictive analytics with mathematical optimization to determine the most effective customer outreach strategy.

By integrating churn probabilities, customer lifetime value, and action costs, the model identifies which customers to target and how to allocate the weekly retention budget for maximum financial impact.

The baseline run demonstrated that the model efficiently utilized the \$150 budget, treating 75 customers and generating a net value of \$3,478.82 with an ROI exceeding 2,300 percent. Treated customers were concentrated among high-risk and high-value users, especially within the Premium and Family segments. All subscription segments met the required 15 percent fairness floor, confirming equitable treatment across the customer base. Personalized email emerged as the most cost-effective channel in this setup.

Sensitivity analysis showed that while increasing the weekly budget leads to higher retained value, the rate of improvement declines sharply beyond \$400 to \$500. The results indicate that an optimal spending range of \$250 to \$400 per week offers the best trade-off between cost and retention performance.

Overall, the optimization model proved effective at prioritizing high-impact customers, respecting operational limits, and ensuring fairness. PlaylistPro can now apply this framework to guide weekly campaign planning, gradually expand its action catalog, and fine-tune predictive inputs through ongoing experimentation. Doing so will help maintain high retention efficiency while maximizing customer lifetime value in a sustainable and data-driven way.