Project 2 Report - Dynamic Programming for Airline Overbooking

Optimization II

Prof. Daniel Mitchell

Team: Biagio Alessandrello (bja2372), Jenna Ferguson (jf38533),
Kimble Horsak (kgh632), Twinkle Panda (tp26545)

April 6th, 2025

Table of Contents

Problem Overview

Airline overbooking has been a long-standing practice in the aviation industry. Historically, airlines have oversold seats to mitigate losses caused by passengers who book but fail to show up. This strategy is rooted in maximizing the utilization of limited flight capacity, especially given the thin margins under which most airlines operate. While overbooking increases the probability of a full flight, it also introduces the potential for denied boarding, passenger dissatisfaction, and compensation costs — all of which airlines must carefully manage.

Most airlines today use historical no-show rates and demand patterns to inform their overbooking policies. They often rely on rules-based systems or simple heuristics to decide how many tickets to sell beyond seat capacity. However, these approaches may not be optimized to account for evolving pricing strategies, demand fluctuations, and real-time sales opportunities. As a result, there is a growing interest in using data-driven techniques, like dynamic programming and simulation, to guide overbooking and pricing decisions more effectively. This is not just relevant for airline ticketing, but can also be applied across industries where perishable inventory is sold in advance — such as hotel reservations, event ticketing, or rental car bookings. In each case, providers aim to maximize revenue while minimizing customer dissatisfaction and penalty costs.

Restating the Problem

This project addresses the question: **"Should an airline overbook coach seats, and if so, by how much?"** We approach this by modeling an airline's ticket sales process over a 365-day period before departure. Each day, the airline chooses optimal prices for coach and first-class tickets. Ticket sales follow probabilistic demand curves based on price and seat class, and passengers show up for their flights with known probabilities. The challenge is to determine how many extra coach tickets the airline should allow itself to sell — and at what daily prices — in order to maximize expected discounted profit.

The Business Challenge

The core business trade-off lies between **maximizing revenue through aggressive ticket sales** and **minimizing overbooking costs**. These costs include compensating coach passengers who are bumped to first class, and in worse cases, those who must be denied boarding entirely. While offering more tickets may improve revenue, the risk of too many passengers showing up — exceeding seat capacity — can erode those gains through costly reimbursements and reputational damage. The solution must strike a careful balance between revenue opportunity and operational risk.

Constraints and Project Versions

Several constraints define the problem space:

- First-class tickets cannot be oversold.
- Coach tickets can be oversold, but only up to a defined overbooking policy (e.g., 5-15 seats).
- A passenger can only book one seat per day, with demand and show-up probabilities known.
- Demand increases over time to simulate seasonality.
- Passengers who would have bought first-class seats will not settle for coach if first-class is unavailable.

We tackle multiple versions of the problem:

1. **Baseline overbooking policy** with a 5-seat oversell limit.
2. **Iterative exploration** of policies overselling coach tickets by 6 to 15 seats.
3. **Flexible pricing policy** that allows the airline to skip coach sales on certain days.
4. **Seasonality model**, adjusting demand probabilities over time.
5. **Forward simulation** to evaluate the real-world impact of each policy across scenarios.

Each version builds on the last to form a comprehensive analysis of overbooking strategies, helping the airline make informed, profitable decisions.

Methodology

Several key parameters define the core environment of the airline ticket pricing problem and remain constant across all scenarios analyzed in this project. These assumptions shape the structure of the dynamic programming model and allow for fair comparison between policies. First, There are 365 **days until the plane departs**, meaning the airline has 365 daily opportunities to sell tickets. On the plane, there are 100 **coach seats**, and 20 **first class seats**, these are fixed capacities that define the physical constraints of the aircraft. Our strategy considers only two **price points** for each class: $300 and $350 for coach, and $425 and $500 for first class. If the plane is overbooked, we first bump the **excess passengers**

into the remaining seats in first class, which costs $50. When first class is full, we **bump passengers off the plane**, which costs us $425. We discount future revenues so that a dollar earned one year from now is worth 17% less than a dollar earned today. This annual rate is converted into a **daily discount factor** to account for the value of profits received on each of the 365 days leading up to the flight.

$$\text{Discount Factor} = \frac{1}{1 + \frac{0.17}{365}}$$

We have chosen to tackle these problems using dynamic programming. Dynamic programming is a powerful method of solving sequential decision-making problems where the outcome of current decisions influences future opportunities. Every dynamic programming solution involves a 6 essential components that define the structure of the problem and guide how the optimal policy is computed: state variables, decision variables, dynamics, a value function, a formulated bellman equation, and a clear terminal condition

State Variables

State variables describe the current status of the system, and include all the information necessary to make a decision at any point in time. They allow the model to track how far along we are in the process, what resources have been used, and what's still available. In all of our scenarios, the state variables consist of the current day, the number of coach tickets sold, and the number of first-class tickets sold, providing a complete snapshot of the system at any point in time.

Decision Variables

Decision variables are the actions we can take in each state. The goal of dynamic programming is to find the best action to take in each possible state, so defining the available choices is essential. Our decision variables vary across our different scenarios.

Dynamics

Our dynamics describe how the state changes over time based on the decisions made and the randomness in the system. They capture how the system evolves, from today's state to tomorrow's, depending on the action taken and the level of uncertainty involved. When the airline sets prices, there's a probability that a ticket is sold. If a ticket sells, the number of tickets sold increases by one. The number of days remaining always decreases by one. These dynamics determine how the system transitions to new states.

Value Function - Dynamic Programming Setup

The value function stores the maximum expected reward (ie profit) we can earn from a given state onward. It captures the long-term benefit of any state, allowing the model to compare short-term rewards with long-term gains. For each possible number of tickets sold and each remaining day, the value function stores the best expected discounted profit the airline can earn from that point until the day of the flight.

Bellman Equation

The Bellman equation defines the recursive relationship used to compute the value function. It says that the value of an action today is equal to its immediate payoff plus the discounted value of the best future state. This equation is what allows our model to build optimal decisions backward from the final time step. The best action that we determine is the one that maximizes this total value.

Terminal Condition

This is the value of the system at the final time step, when no more decisions can be made because no more tickets can be sold - the plane is in the air. Dynamic programming must start with known outcomes at the end of the decision horizon in order to work backwards. On the day of the flight, we incur overbooking costs if more passengers show up than there are seats. This cost becomes the terminal value in the model. For all scenarios discussed in our project, the terminal condition remains constant as the expected cost of overbooking on the day of the flight.

Even though a certain number of tickets have been sold, we are uncertain about the number of passengers who will actually show up on the day of the flight. We tackle this with a probabilistic approach. We know that a coach ticket holder shows up with a 95% chance, and a first-class ticket holder with a 97% chance. Using a binomial distribution, we compute the likelihood of every possible number of passengers showing up in each class. For each possible combination of coach and first-class passengers who might show up, based on how many tickets were sold, we calculate the cost of overbooking: if more coach passengers show up than there are coach seats, some are either bumped to first-class (if there's room) or denied boarding, both of which incur penalties. By multiplying each scenario's cost by its probability and summing across all possibilities, we derive the expected overbooking cost. This value becomes the terminal entry in our dynamic programming model, since no more decisions can be made after this point, only costs are realized.

$$\text{ExpectedCost} = \sum_{\text{coach\_show}=0}^{\text{coach\_sold}} \sum_{\text{fc\_show}=0}^{\text{fc\_sold}} P_{\text{coach}}(\text{coach\_show}) \cdot P_{\text{fc}}(\text{fc\_show}) \cdot \text{ScenarioCost}(\text{coach\_show}, \text{fc\_show})$$

The formula above calculates the expected overbooking cost for a given combination of coach and first-class tickets sold. Since this cost is incurred on the day of the flight and does not generate any revenue, we store it as a negative value in our dynamic programming matrix V, which tracks the maximum expected profit from each possible state. This cost becomes the terminal value in the model and serves as the foundation for working backward through time. The logic implemented in our code for all scenarios is shown below:

```
def calculate_terminal_costs(V, max_coach_tickets, first_class_seats, coach_show_prob,
                             first_class_show_prob, coach_seats, cost_bump_to_fc, cost_bump_off):
  for coach_sold in range(max_coach_tickets + 1): # loop through all possible coach tix sold
      for fc_sold in range(first_class_seats + 1): # loop through all possible fc tix sold
          cost = 0
          for coach_show in range(coach_sold + 1): # through coach passengers who might show up
              c_prob = binom.pmf(coach_show, coach_sold, coach_show_prob)# prob of this scenario
              for fc_show in range(fc_sold + 1): #loop through fc passengers who might show up
                  fc_prob = binom.pmf(fc_show, fc_sold, first_class_show_prob)# prob of this
                  scenario_prob = c_prob * fc_prob # multiply bc independent events
                  if coach_show > coach_seats: # if more people show up than capacity
                      bumped = coach_show - coach_seats # how many people can't make it in coach
                      available_fc = max(0, first_class_seats - fc_show) # fill up fc with bumps
                      bumped_to_fc = min(bumped, available_fc) # fill up fc with bumps
                      bumped_off = bumped - bumped_to_fc # bump the rest off the plane
                      cost += scenario_prob * (bumped_to_fc * cost_bump_to_fc + bumped_off * cost_bump_off)
          V[coach_sold, fc_sold, -1] = -cost #store as terminal value at day 365 in value matrix
  return V
```

Overbooking Policy Evaluation (Steps 1–2)

Since it is not 100% certain that all passengers who bought a ticket will show up, we can choose to sell more tickets than there are seats on the plane. This incurs costs, and we assess which overbooking policy leads to the largest profits for us, while determining the optimal price to charge for coach and first-class tickets on each day leading up to a flight. The goal is to maximize the airline's total expected discounted profit, taking into account uncertain ticket sales and overbooking penalties. In this model, we assume fixed sale probabilities at each price point, independent ticket purchase and show-up decisions across passengers, and constant overbooking penalties. The model also assumes no customer cancellations or refunds outside of no-shows.

To solve this, we compute the maximum expected profit at every possible state: defined by the number of tickets sold in each class and the number of days remaining. At each state, the DP algorithm evaluates all possible price combinations and calculates the expected immediate revenue along with the discounted future profit, based on how likely it is that tickets will be sold. This process continues in reverse chronological order, starting from the day of the flight (where only overbooking costs are incurred) and working backward to day 0. Overbooking costs are computed using the binomial distribution to model how many passengers actually show up. Lets dive deeper into the logistics of this methodology.

Choice Variables

In this model, our choice variables represent the decisions the airline makes each day about what ticket prices to offer in coach and first class. Specifically, in each state, when both cabins have availability, there are two choices to be made: the price of coach tickets (high or low) and the price of first-class tickets (high or low). These pricing choices affect the probabilities of selling a ticket in each cabin, and therefore impact both immediate revenue and the future value of the state. When first class is sold out, the choice is limited to selecting only a coach price (and the first-class action is automatically set to high price but unused).

Value Function

The value function V(t,c,f) captures the maximum expected discounted profit from day t until the final day T, given that c coach tickets and f first-class tickets have been sold. At each decision point, we choose a pricing action that maximizes the sum of the immediate expected revenue today plus the discounted expected revenues from all future days. This includes not only tomorrow's outcomes but also the consequences of today's decision on all future states up to the day of departure (time T). The discount factor (γ) ensures that future profits are appropriately weighted, with each additional day into the future receiving a larger discount. In this way, V(t,c,f) fully accounts for both short-term gains and long-term impacts.

$$V(t,c,f) = \max_{\text{action}\in\text{Actions}(c,f)} \begin{array}{l} \text{Expected Revenue Today} \\ \left\{ \begin{array}{l} +\,\gamma \times \text{Expected Future Value at } t+1 \\ +\,\gamma^2 \times \text{Expected Future Value at } t+2 \\ +\cdots \end{array} \right\} \\ +\,\gamma^{T-t} \times \text{Expected Future Value at } T \end{array}$$

Bellman Equation

The bellman equation for this scenario states that the optimal value function V(t,c,f) is the maximum across all possible pricing actions of today's expected profit plus the discounted expected value of future profits. Each pricing action that we choose leads to different probabilities of selling tickets, and affects both immediate revenue and the transition to tomorrow's state. The Bellman equation formally incorporates all four possible pricing combinations for a given state and ensures the pricing strategy which leads to the highest overall expected value across the full selling horizon is chosen. In the equation below, c' and f' are the number of coach seats and first class seats available tomorrow, which is based on what is sold today.

$$V(t,c,f) = \max \begin{Bmatrix} \mathbb{E}\left[\text{Profit}(350,500) + \gamma V(t+1,c',f')\right], \\ \mathbb{E}\left[\text{Profit}(350,425) + \gamma V(t+1,c',f')\right], \\ \mathbb{E}\left[\text{Profit}(300,500) + \gamma V(t+1,c',f')\right], \\ \mathbb{E}\left[\text{Profit}(300,425) + \gamma V(t+1,c',f')\right] \end{Bmatrix}$$

In our code, we apply backward induction to solve for the optimal ticket pricing strategy. Starting from the day of the flight (where only overbooking costs are incurred), we work backwards day-by-day toward the beginning of the selling period. For each possible state (number of coach tickets sold, first-class tickets sold, and days remaining) we evaluate all allowable pricing actions. At each state and day, the algorithm calculates the expected immediate revenue from ticket sales and the discounted expected future value based on transition probabilities (probabilities of different sales outcomes). It then selects the pricing action that maximizes the sum of these two components, storing both the maximum value (V) and the corresponding best action (U). This ensures that at every step, the airline is choosing the ticket pricing policy that leads to the highest possible expected discounted profit from that point forward.

Once we have set up the dynamic programming logic, we determine what actions (pricing strategy) give us the best value (profits), which allows us to determine the expected profit by doing these actions. We then repeat the process while varying the allowed overbooking level from 5 to 15 extra coach tickets. For each overbooking policy, we solve the DP problem to get the expected total discounted profit. We then compare the results to determine the optimal level of overbooking that leads to the highest expected profits.

No-Sale Policy (Part 3)

In Part 3, we build on the framework from the previous scenario but introduce an important new flexibility: the airline can now choose to offer no coach tickets for sale on a given day. In the previous parts, the airline was always forced to either offer a high or low price for coach tickets. However, in this modified strategy, there is a third choice for coach each day: sell at a low price, sell at a high price, or choose not to sell at all (forcing coach demand to zero for that day). This additional option allows the airline to better control overbooking risk by slowing down sales when necessary, especially if many days remain before departure. First-class pricing remains the same, with two options (high or low price) available as long as seats are available. We keep the maximum number of coach seats sold to be 120 (20 seats higher than capacity). We solve a new dynamic programming problem that accounts for this expanded action space and then compare whether having the ability to "pause" coach sales improves or worsens the airline's expected discounted profit compared to the best policy from Part 2.

Choice Variables

In Part 3, the airline's choice variables at each state are expanded. For coach tickets, the airline now has three options each day: (1) do not sell any coach tickets (no-sale option), (2) sell at a low price ($300), or (3) sell at a high price ($350). For first-class tickets, the airline still chooses between two options: (1) sell at a low price ($425) or (2) sell at a high price ($500).
Each action is a combination of a coach and a first-class decision. We store each combination of actions in a matrix similar to the following:

Coach Option	First Class Option	Action Index
No Sale	Low Price	0
No Sale	High Price	1
Low Price	Low Price	2
Low Price	High Price	3
High Price	Low Price	4
High Price	High Price	5

Bellman Equation

Our Bellman equation for part 3 now includes the consideration of the expected profit when for all 6 actions:

$$V(t,c,f)=\max\begin{Bmatrix}\mathbb{E}\left[\text{Profit}(\text{No Coach Sale, Low FC Price})+\gamma V(t+1,\text{new }c,\text{new }f)\right],\\ \mathbb{E}\left[\text{Profit}(\text{No Coach Sale, High FC Price})+\gamma V(t+1,\text{new }c,\text{new }f)\right],\\ \mathbb{E}\left[\text{Profit}(\text{Low Coach Price, Low FC Price})+\gamma V(t+1,\text{new }c,\text{new }f)\right],\\ \mathbb{E}\left[\text{Profit}(\text{Low Coach Price, High FC Price})+\gamma V(t+1,\text{new }c,\text{new }f)\right],\\ \mathbb{E}\left[\text{Profit}(\text{High Coach Price, Low FC Price})+\gamma V(t+1,\text{new }c,\text{new }f)\right],\\ \mathbb{E}\left[\text{Profit}(\text{High Coach Price, High FC Price})+\gamma V(t+1,\text{new }c,\text{new }f)\right]\end{Bmatrix}$$

When first-class is fully sold out, the airline can no longer sell additional first-class tickets. In this case, we restrict the available actions to only those where the first-class price is automatically set to the high price, even though no further sales can occur. The only decision left is which coach price to set: no sale, low price, or high price. On the other hand, if first-class seats are still available, the airline has full flexibility to choose any combination of coach and first-class pricing. This means there are six total action options each day: choosing whether to sell coach at no sale, low, or high price, combined with a low or high price for first-class.

After solving the dynamic programming model with the no-sale option in Part 3, we compare the resulting expected discounted profit to the best overbooking policy identified in Part 2. This comparison allows us to assess whether giving the airline the additional flexibility to choose "no sale" on a given day improves or worsens overall profitability. By evaluating the profits side by side, we can determine if the ability to pause coach ticket sales strategically leads to better outcomes than simply managing overbooking levels without a no-sale option.

Time-Varying Demand (Step 4)

In Part 4, we extend the model by introducing a realistic seasonality component to ticket demand. As the day of departure approaches, customers are more likely to purchase tickets. To capture this effect, we adjust the sale probabilities for both coach and first-class tickets by a time-dependent factor on each day, given by 0.75 + t/730. Early in the selling period, demand is lower, and as we get closer to departure, demand rises.

$$\text{Seasonality Factor}=0.75+\frac{t}{730}$$

The structure of the decision-making remains the same as in Part 3, where we choose each day whether to sell no coach tickets, offer coach at a low or high price, and whether to offer first-class at a low or high price. However, because the effective probabilities of selling tickets now vary each day, the expected immediate revenue and transition probabilities change dynamically over time. This addition makes the pricing strategy more sophisticated, as we must balance between early sales at lower probabilities and the opportunity for stronger demand later in the selling horizon. We keep the amount of coach tickets sold as a value of 120.

The Bellman equation retains the same fundamental form, maximizing the sum of today's expected profit and the discounted future value, but now with sale probabilities that depend on the current day. This seasonal adjustment allows us to model a more realistic airline sales process, where timing plays a crucial role in revenue optimization.

Simulation (Step 5)

After solving the dynamic programming problem and determining the optimal ticket pricing policy (stored in the U matrix), we use forward simulation to evaluate how this policy would perform in practice under real-world uncertainty. In the simulation, randomness is introduced at two key stages: whether a ticket sale occurs each day, and whether a passenger shows up for the flight on departure day. By simulating thousands of flight sales over the course of a year, we can estimate important performance metrics such as average discounted profit, profit volatility, the rate of overbooking, and the cost of bumping passengers. This allows us to realistically assess the effectiveness and risks of our policy beyond the deterministic world of dynamic programming.

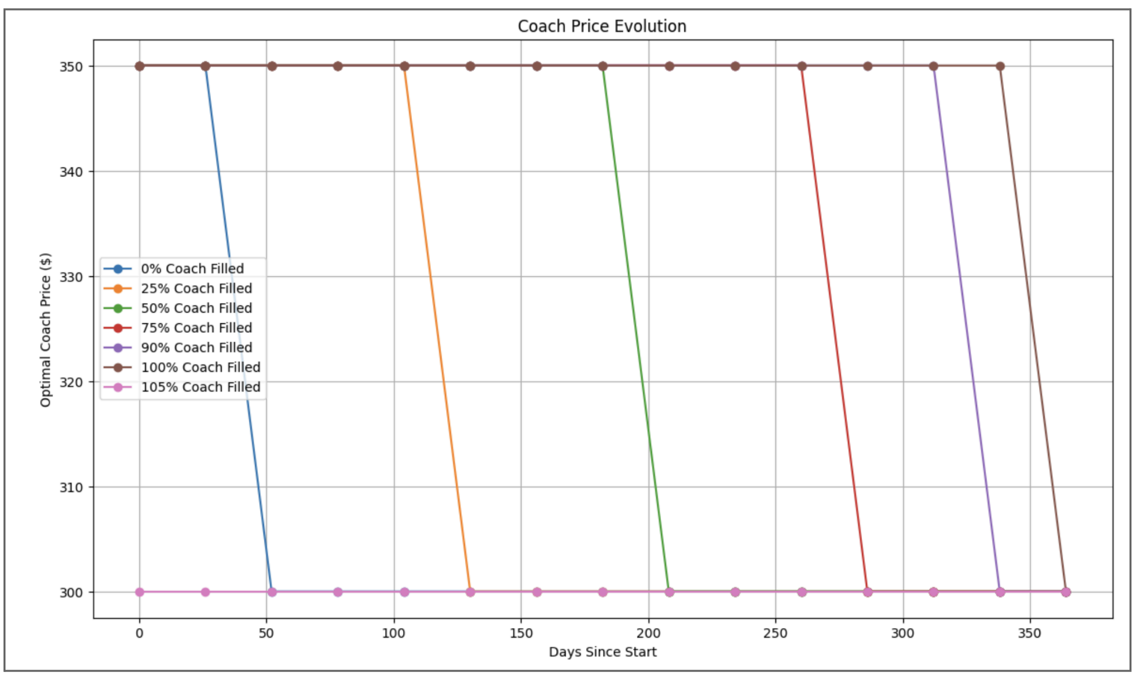
Each simulation run represents one full flight sales cycle over 365 days. On each day, based on the optimal policy, the model chooses ticket prices for coach and first class, adjusts sale probabilities for seasonality, and uses random draws to determine whether a sale occurs. Revenue is collected day-by-day with appropriate discounting, and ticket sales accumulate until the day of departure. On the departure day, passenger show-ups are simulated using binomial distributions, and if overbooking occurs, bumping costs are incurred. The logic is similar to how the terminal condition was calculated. The final profit for the flight is calculated as total discounted revenue minus any overbooking penalties, and the results are aggregated across all simulated flights to provide performance statistics.

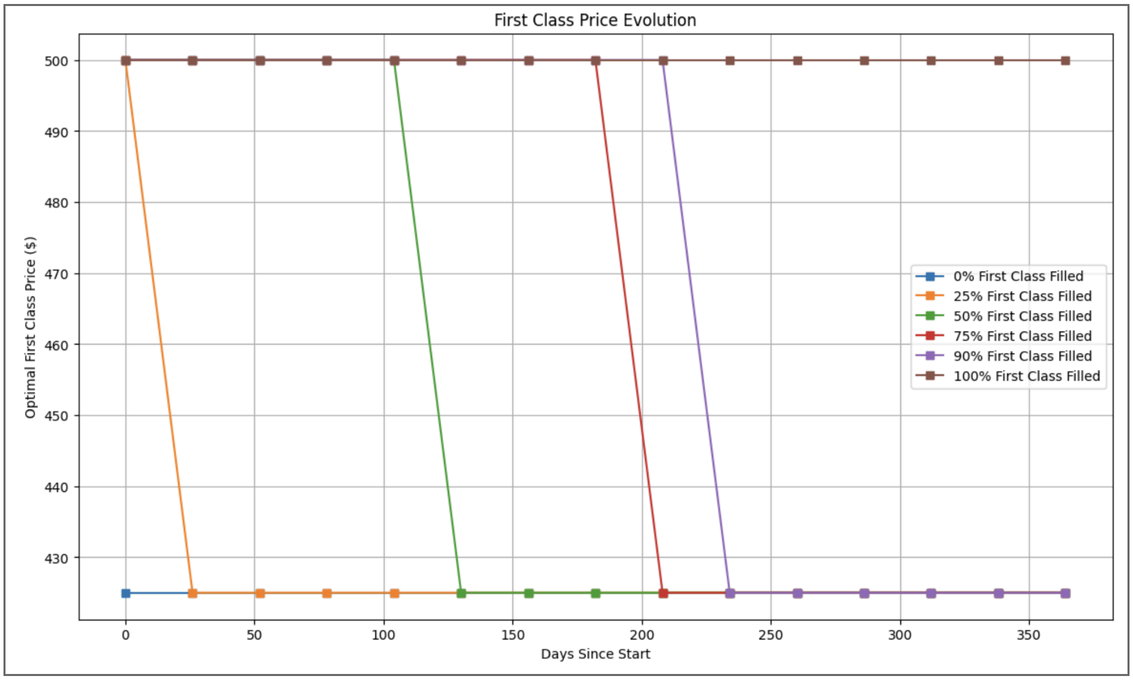
Results

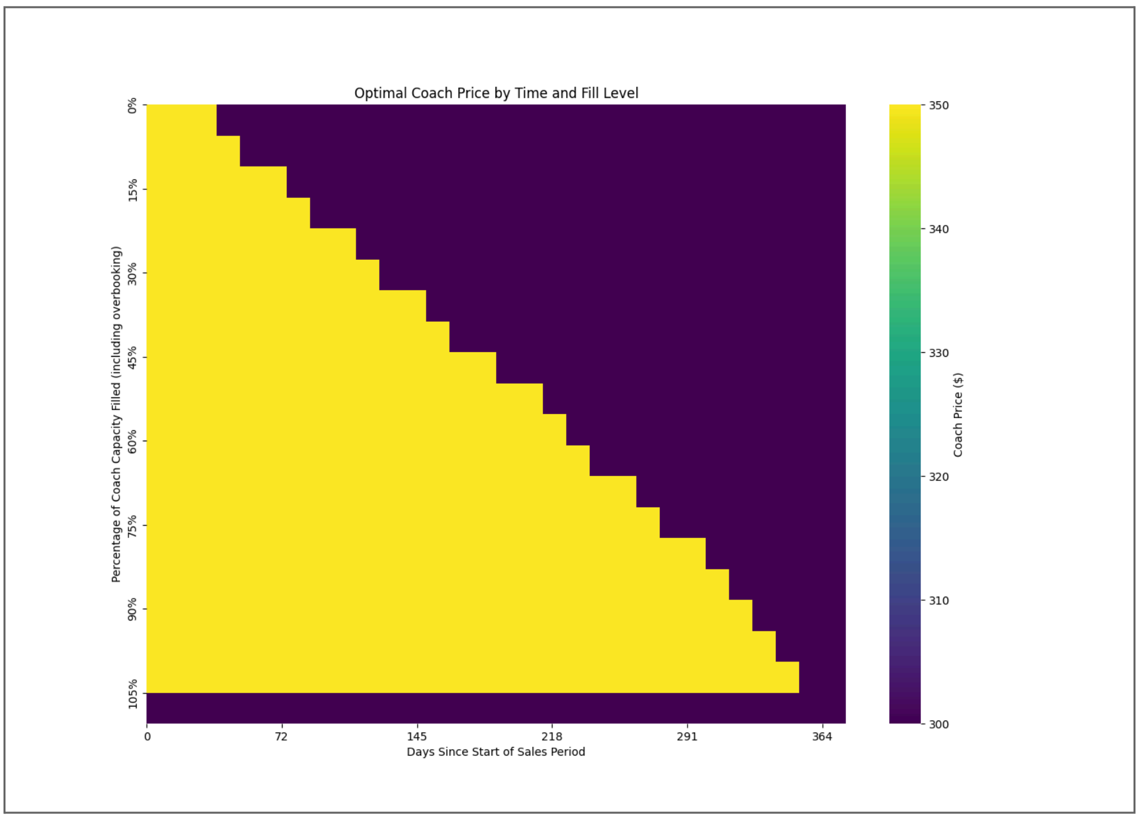
We utilized the methodology explained above to maximize the airline's expected discounted profit while managing overbooking risks and costs.

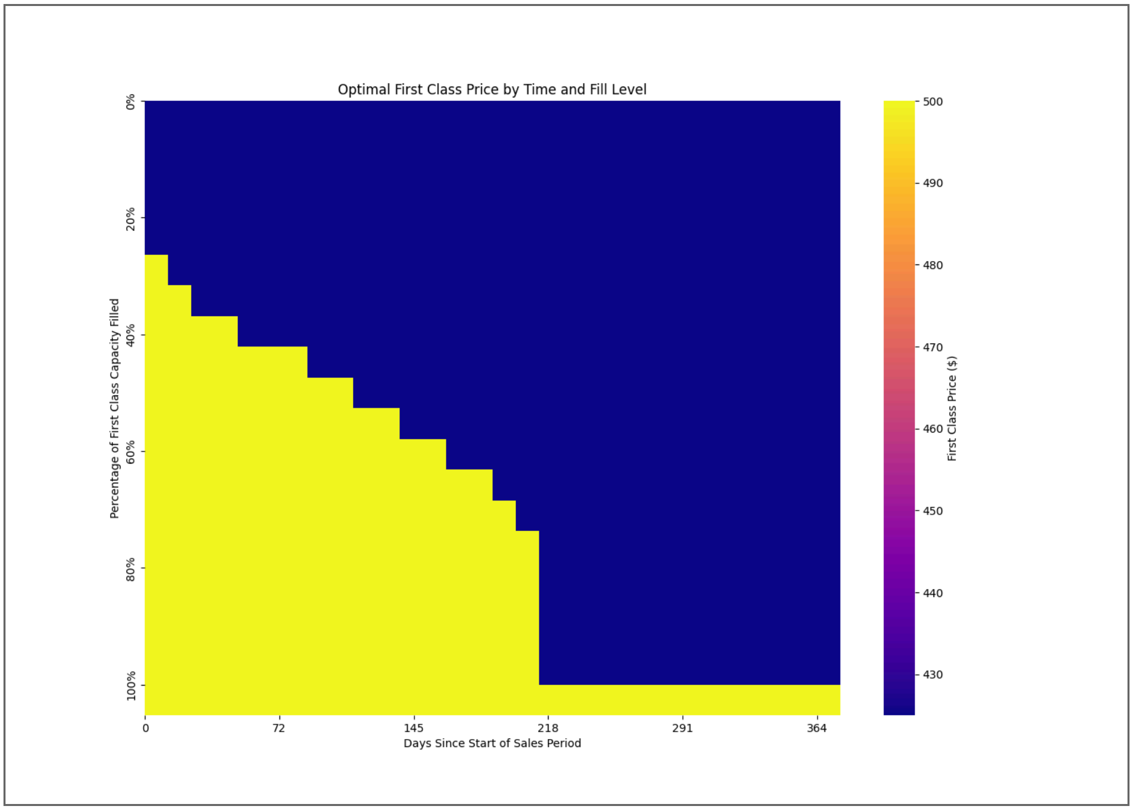
Overbooking by 5 Seats

With an allowed overbooking of 5 coach seats, the expected discounted profit was computed using a dynamic programming model that included two-tier pricing for coach and first-class tickets, probabilistic customer show-up rates while considering overbooking penalties of $50 for upgrades to first-class and $425 for denied boarding. This resulted in an expected discounted profit of **$41,886.16**. Optimal pricing strategies involved initially setting higher ticket prices and strategically reducing prices as the flight date approached, especially as seat occupancy increased.




First Class Price Evolution
Optimal First Class Price ($)
Days Since Start
0% First Class Filled
25% First Class Filled
50% First Class Filled
75% First Class Filled
90% First Class Filled
100% First Class Filled


Optimal Coach Price by Time and Fill Level
Percentage of Coach Capacity Filled (including overbooking)
Days Since Start of Sales Period
Coach Price ($)

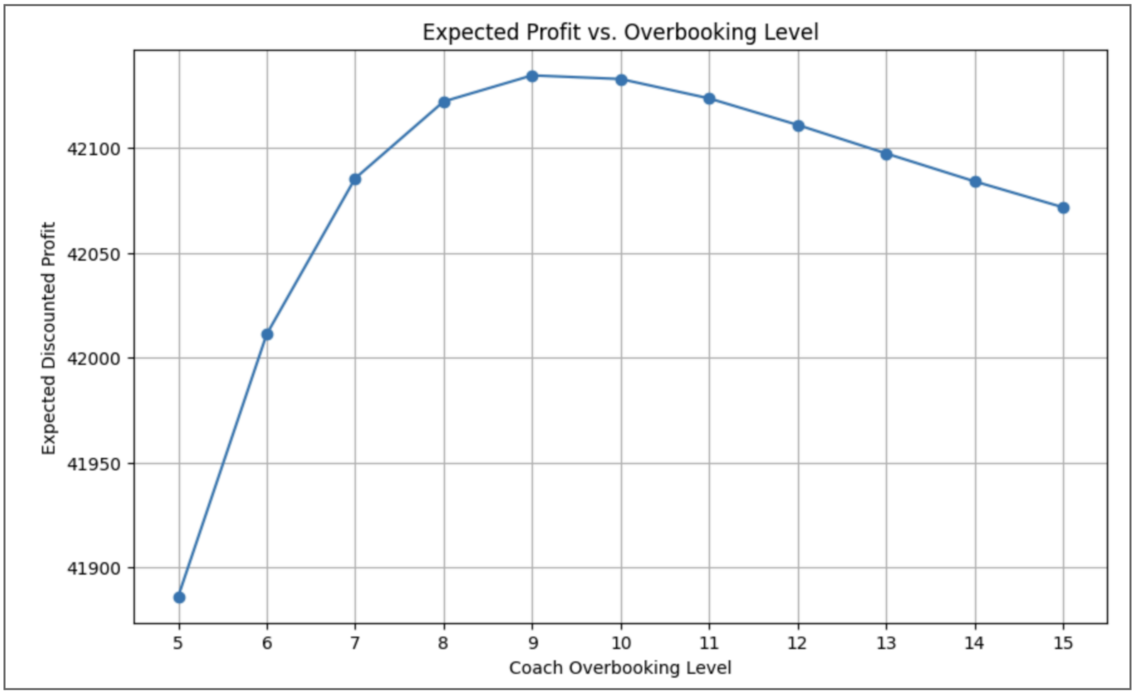


Best Overbooking Policy

We expanded the dynamic programming solution by evaluating overbooking allowances ranging from 6 to 15 additional coach seats, finding that the optimal overbooking level is **9 seats**, yielding the highest expected discounted profit of **$42,134.62**. Profit steadily increases up to this optimal point and then starts to decline, indicating diminishing returns as the risk of overbooking costs grows. Thus, the 9-seat overbooking level effectively balances increased revenue against potential overbooking costs.

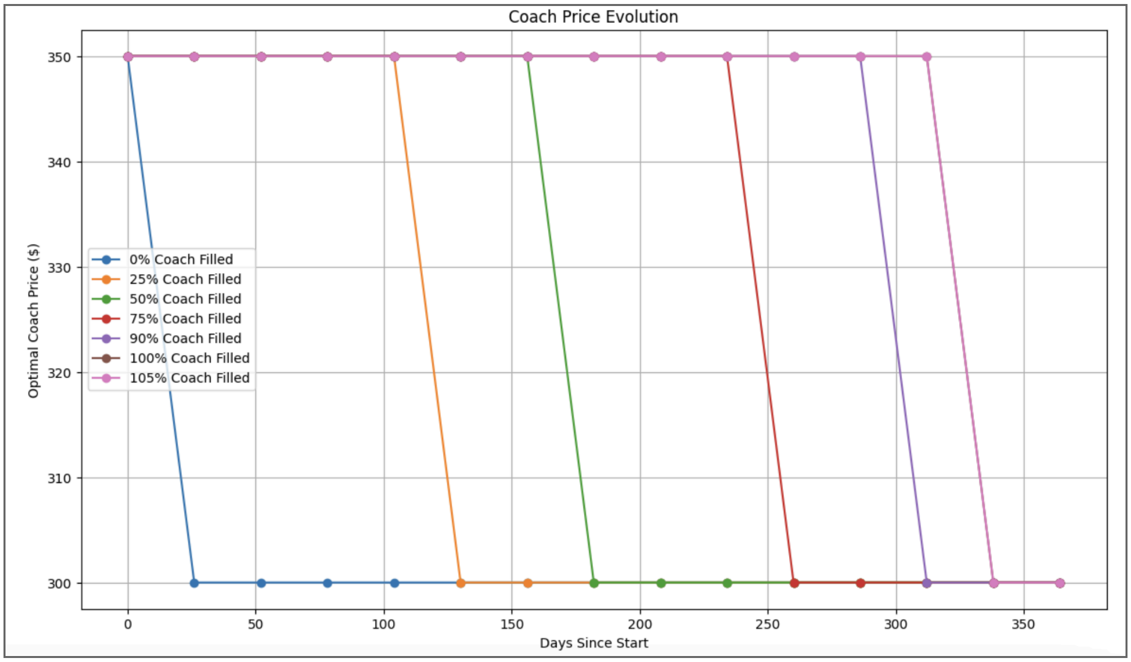
Overbooking Level	Expected Profit	Overbooking Level	Expected Profit
6	$ 42011.22	11	$ 42123.67
7	$ 42085.54	12	$ 42111.03
8	$ 42122.17	13	$ 42097.42
9	**$ 42134.62**	14	$ 42084.11

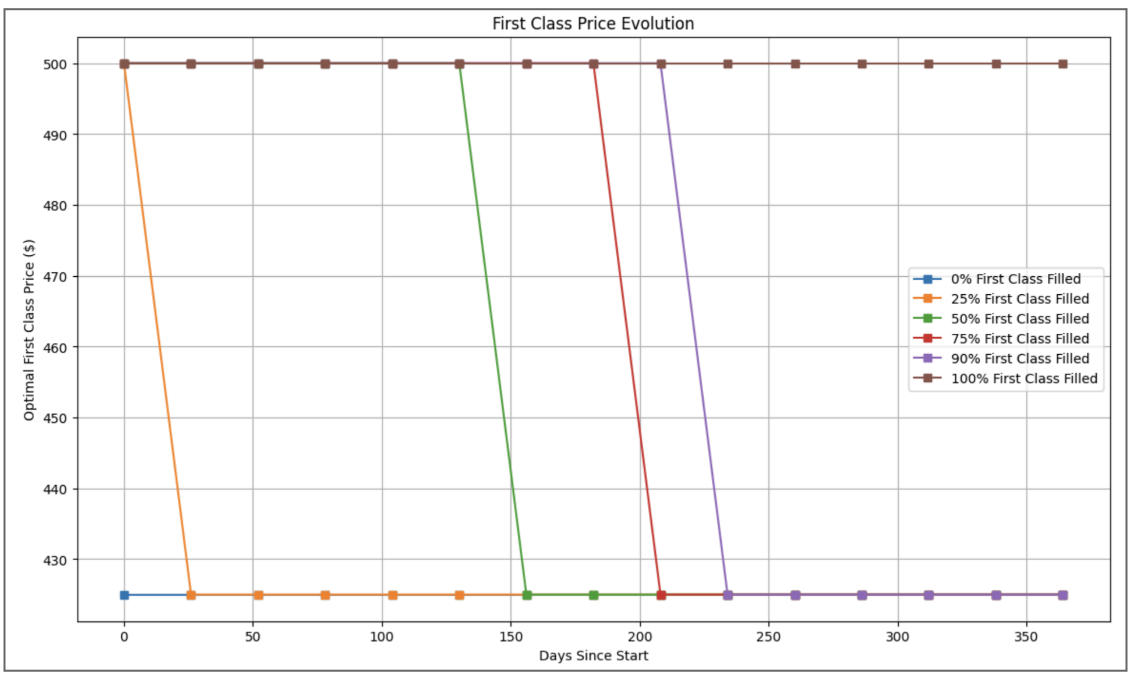
10	$ 42132.90	15	$ 42071.74



No-Sale Option Comparison

The introduction of a "no-sale" option, allowing the airline to stop coach ticket sales selectively, led to an even higher expected profit of **$42,139.89**. Pricing strategies showed increased flexibility, delaying price reductions significantly until later in the booking period, thus optimizing revenue and controlling overbooking risks more effectively. This strategic flexibility proved superior to fixed-cap approaches.

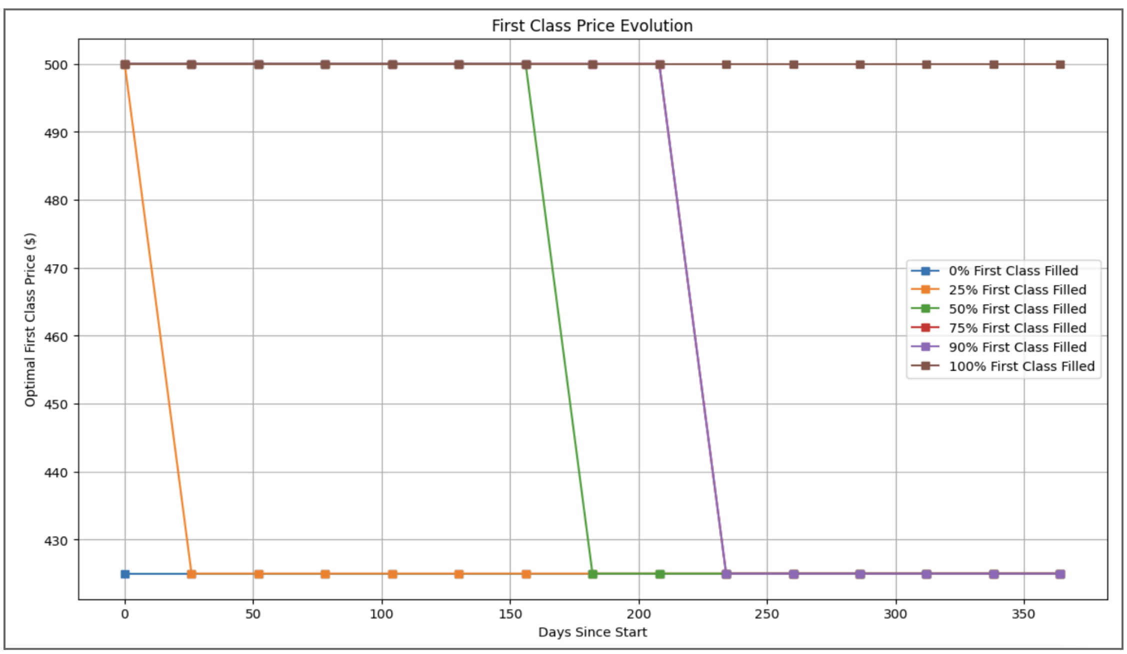




Demand Seasonality Adjustment

To reflect realistic passenger demand trends, we adjusted daily sale probabilities to increase as departure approached. This adjustment enhanced the model's realism but reduced expected profit slightly to

$41,830.46. Despite the lower profit relative to non-seasonal scenarios, this approach realistically accounted for typical airline industry dynamics and reflected genuine market conditions.



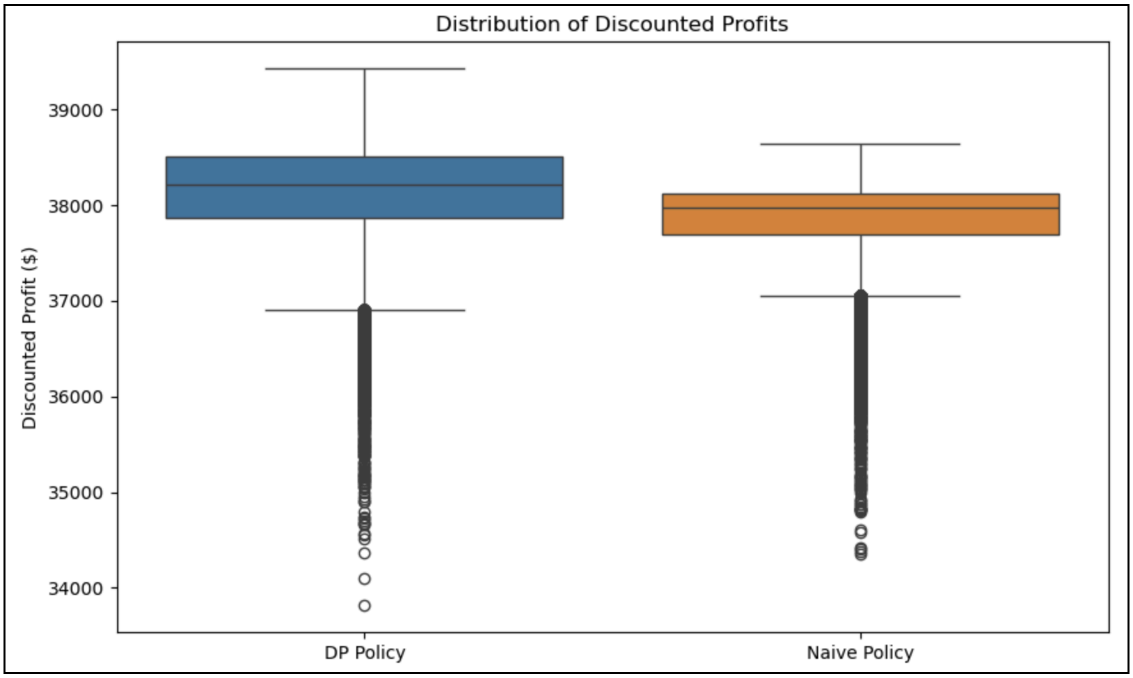
Simulation Insights

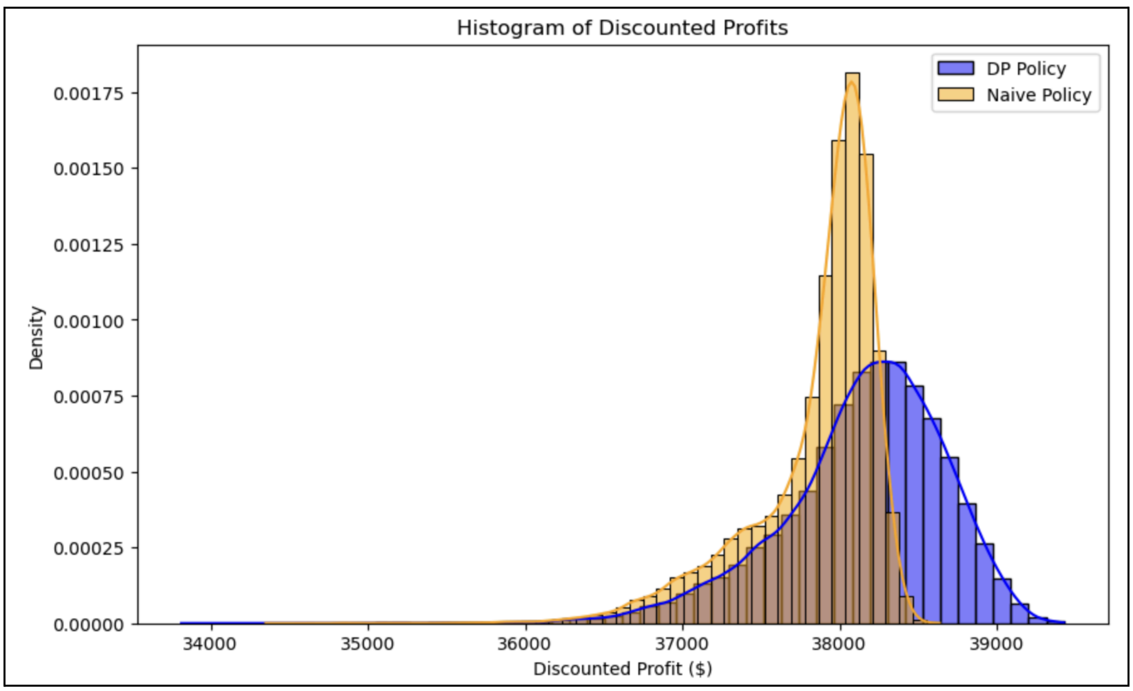
Forward simulations were conducted to compare the performance of dynamic programming (DP) policies: ‘Overbooking Policy - 5 seats’, ‘Overbooking Policy - 9 seats’, ‘No Sale Option Policy’ and ‘Seasonality Inclusion Policy’. All of these approaches were compared against a baseline naive approach.

Overbooking Policy - 5 seats

Forward simulation results for the overbooking policy allowing 5 extra coach seats indicate that the DP optimized strategy yields an expected profit of **$38,148.03**, with a profit volatility of **$527.83**. The simulation shows that 39.27% of cases are overbooked and 28.54% of simulations result in passengers being kicked off, with an average overbooking cost of **$194.28**. In comparison, the naive policy produces a slightly lower expected profit of **$37,848.68**, similar overbooking percentages (39.29% overbooked and 29.47% kicked off), and an average overbooking cost of **$199.52**. The results are summarized in the table below:

Metric	Naive Policy	DP Policy (5 Seats Overbooking)
Expected Profit	$37,848.68	$38,148.03
Profit Volatility	$411.02	$527.83
Percentage Overbooked	39.29%	39.27%
Percentage Passengers Kicked Off	29.47%	28.54%
Average Overbooking Cost	$199.52	$194.28
Average Coach Tickets Sold	105.00	105.00
Average First-Class Tickets Sold	19.94	19.87

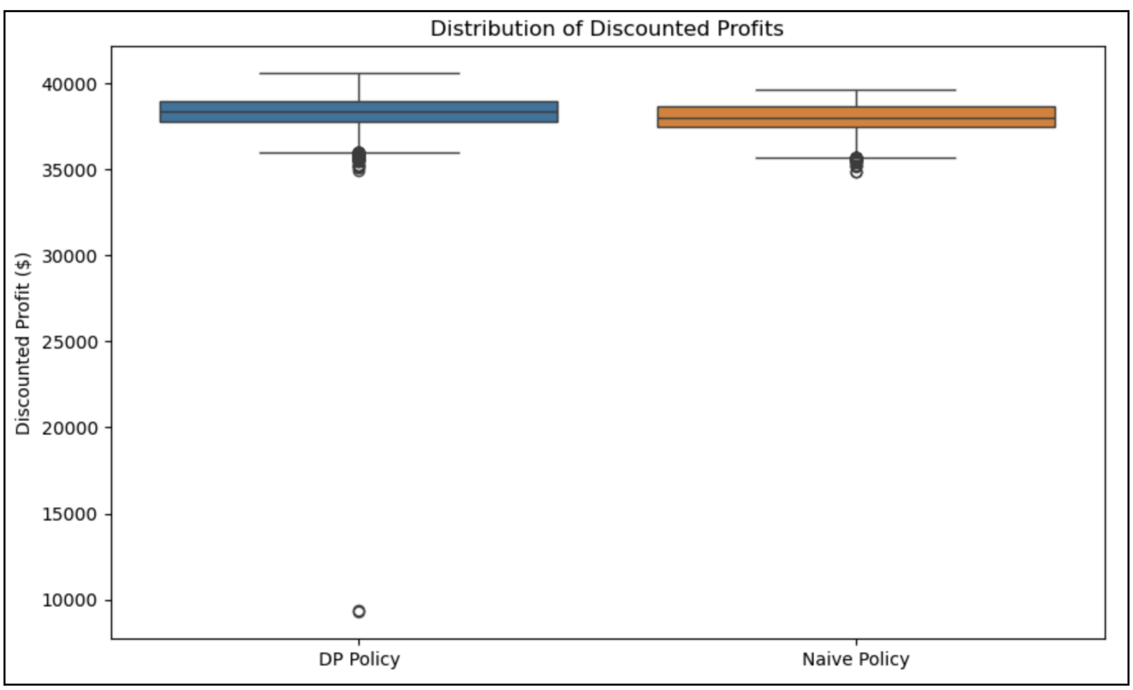


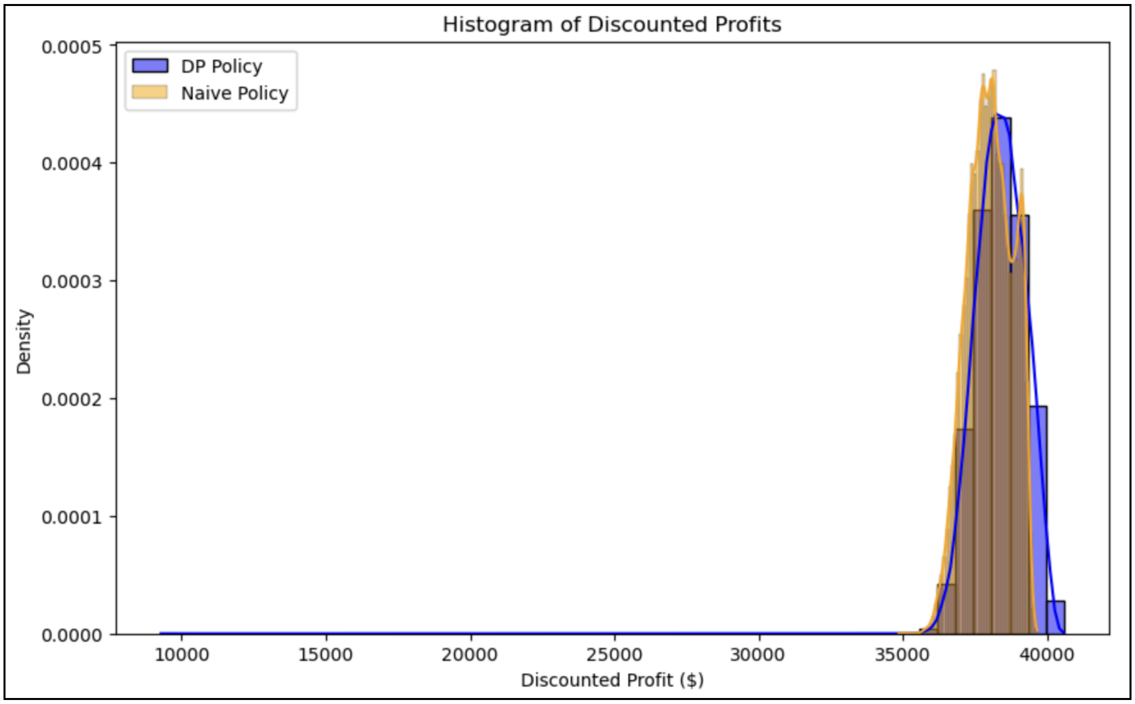


Overbooking Policy - 9 seats

This section evaluates the performance of a Dynamic Programming (DP) optimized policy that allows overbooking by 9 coach seats, compared to a naive overbooking strategy. The DP approach uses intelligent sales control to maximize revenue while managing passenger disruption costs. Despite selling fewer total tickets than the naive policy, the DP policy significantly improves profitability and reduces the frequency and severity of overbooking consequences.

Metric	Naive Policy	DP Policy (9 seats OB)
Expected Profit	$38026.54	$38381.68
Profit Volatility	$767.30	$833.59
Percentage Overbooked	90.43%	90.26%
Percentage Passengers Kicked Off	83.98%	80.85%
Average Overbooking Cost	$1121.35	$1,051.71
Average Coach Tickets Sold	109	109
Average First-Class Tickets Sold	19.94	19.67





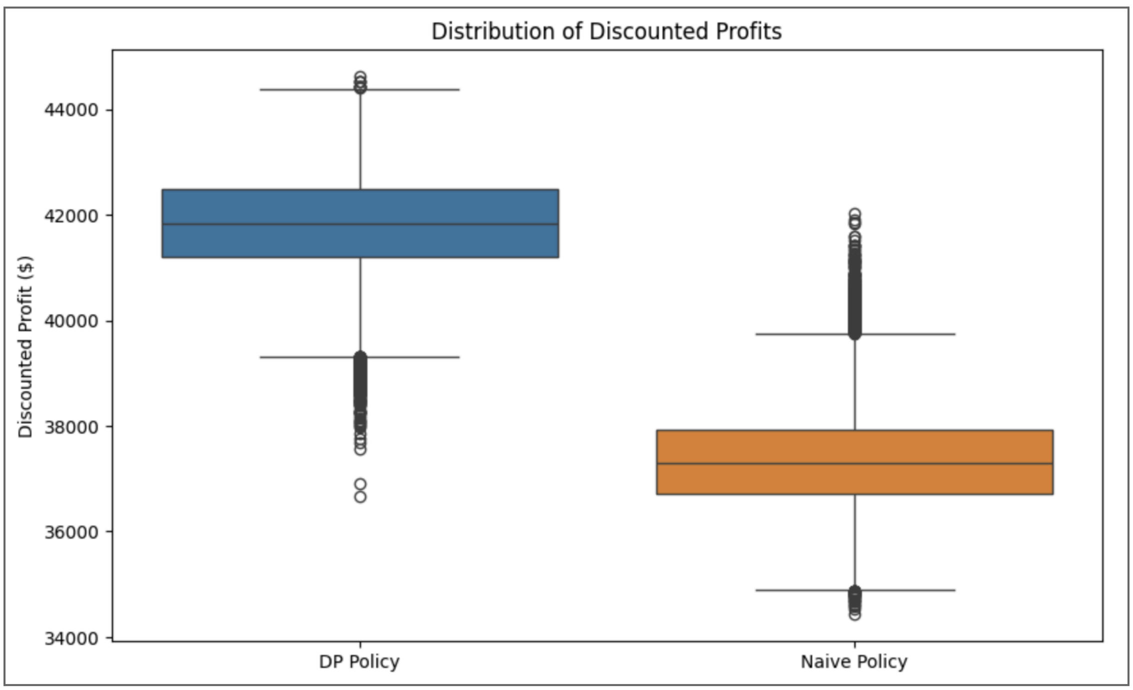
These findings illustrate that while the DP policies for both 5 and 9 additional coach seats lead to improved profitability and reduced overbooking costs compared to the naive approach, the 9-seat overbooking policy yields a higher expected profit even with fewer tickets sold on average. The DP policy delivers a slight uplift in expected profit over the naive approach. It also reduces average overbooking costs by **6%**, highlighting its efficiency in minimizing high-penalty events such as passenger kick-offs.

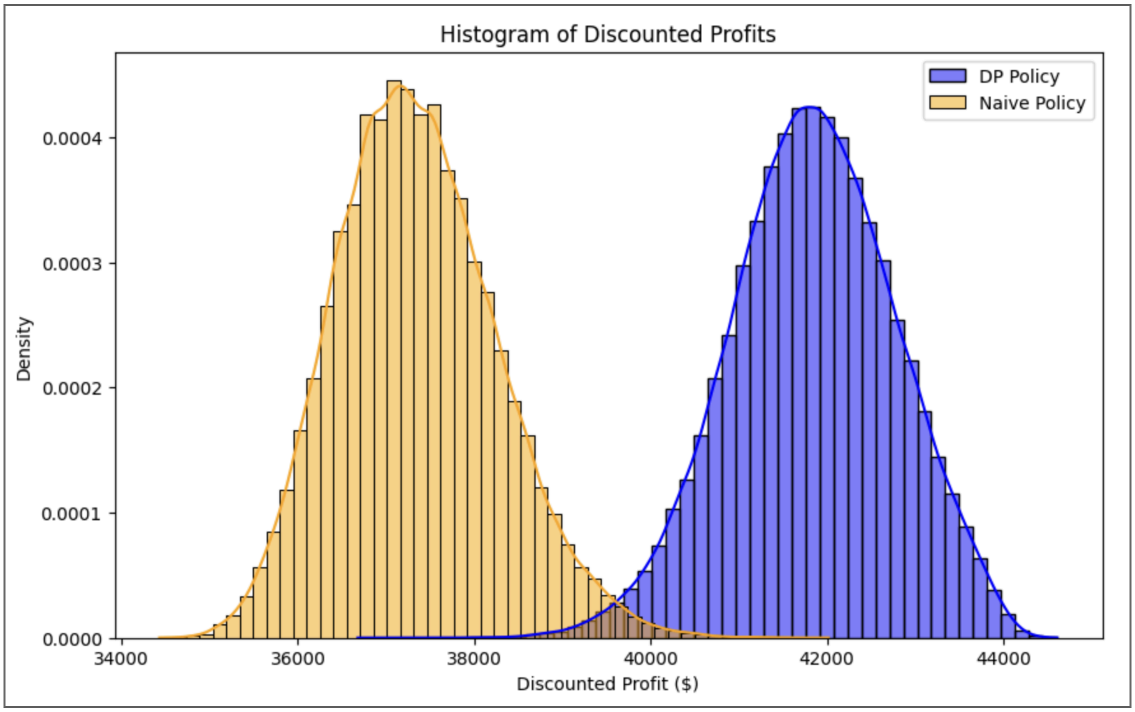
No Sale Option Policy

The DP policy achieved a significantly higher expected profit with an uplift of approximately **11.81%** and lower average overbooking cost compared to the naive policy. It also reduced passenger dissatisfaction by decreasing the frequency of passengers being kicked off flights. These results highlight the importance of strategic, data-driven overbooking and pricing decisions to optimize both revenue and customer experience.

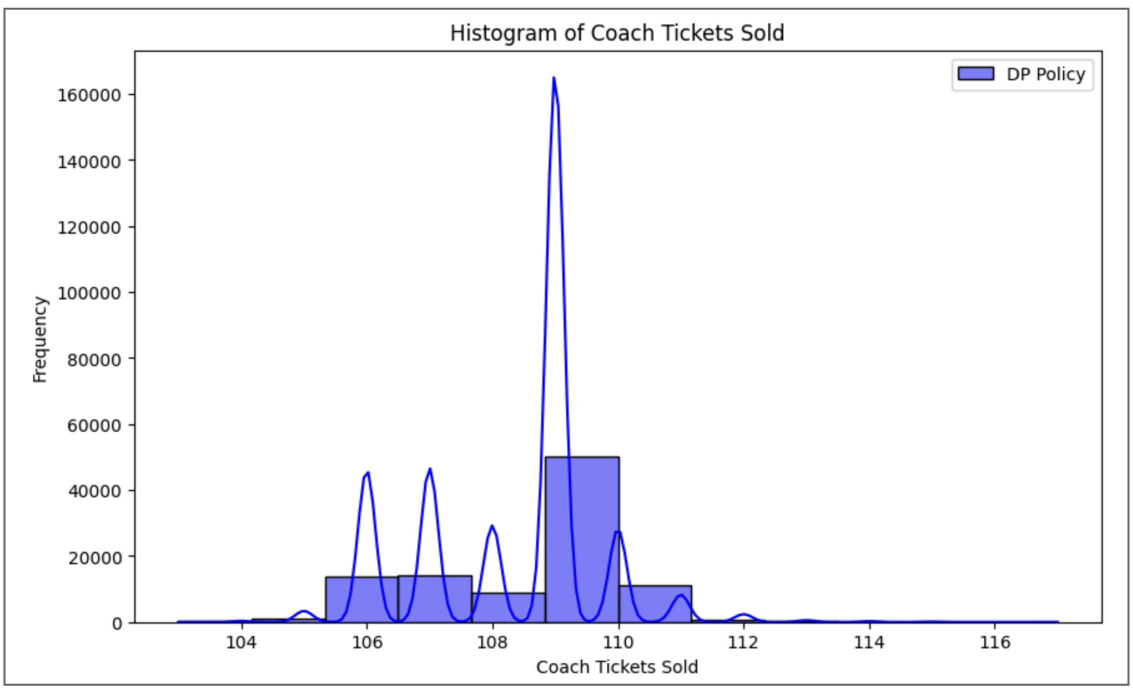
Metric	Naive Policy	No Sale Option
Expected Profit	$37,346.34	$41,759.37
Profit Volatility	$905.69	$937.98

Percentage Overbooked	100.00%	87.43%
Percentage Passengers Kicked Off	100.00%	79.26%
Average Overbooking Cost	$ 4,809.66	$1,020.36
Average Coach Tickets Sold	120.00	108.78
Average First-Class Tickets Sold	19.94	19.75

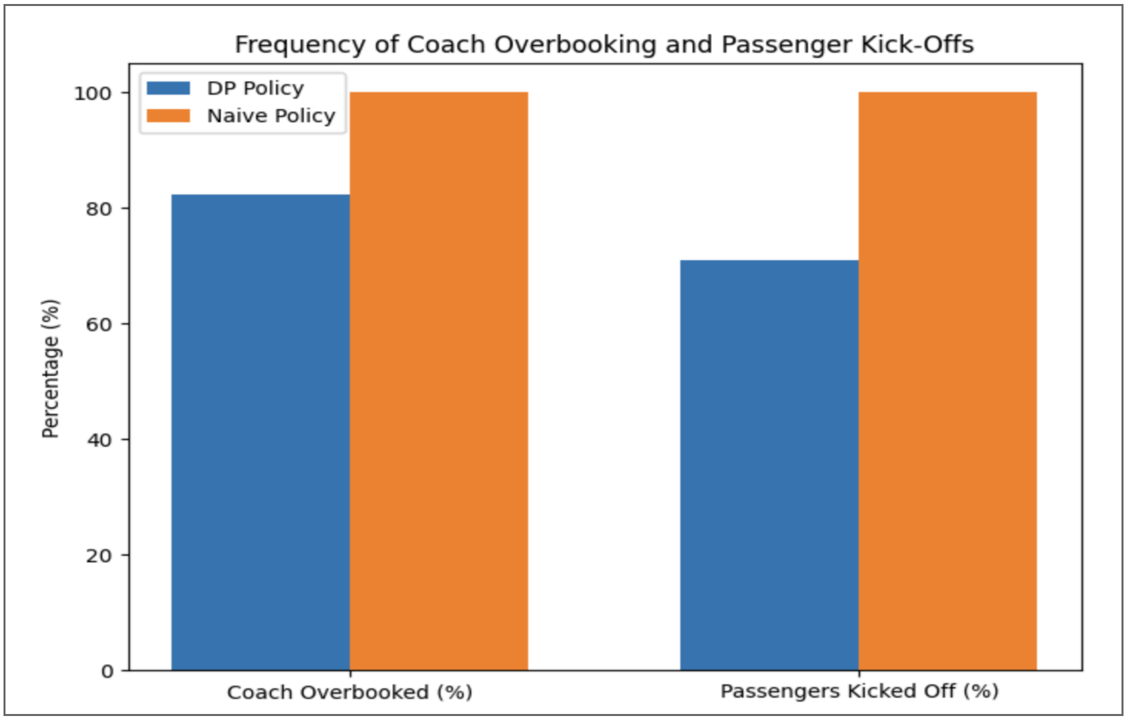




Histogram of Discounted Profits
DP Policy
Naive Policy
Density
0.0000
0.0001
0.0002
0.0003
0.0004
34000
36000
38000
40000
42000
44000
Discounted Profit ($)



Histogram of Coach Tickets Sold
DP Policy
Frequency
0
20000
40000
60000
80000
100000
120000
140000
160000
104
106
108
110
112
114
116
Coach Tickets Sold



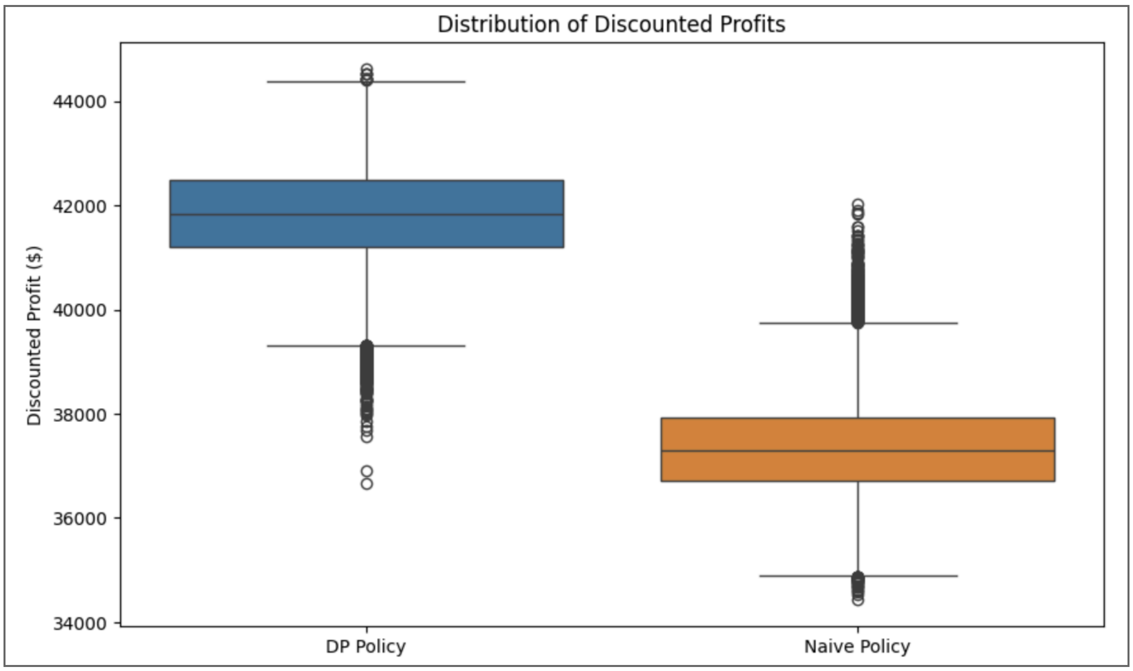
The profit distribution plots reveal that the DP policy not only increased average profit but also maintained reasonable volatility compared to the naive approach. Additionally, the histograms indicate a more controlled distribution of coach tickets sold under the DP policy, reflecting effective management of sales capacity and overbooking risk.

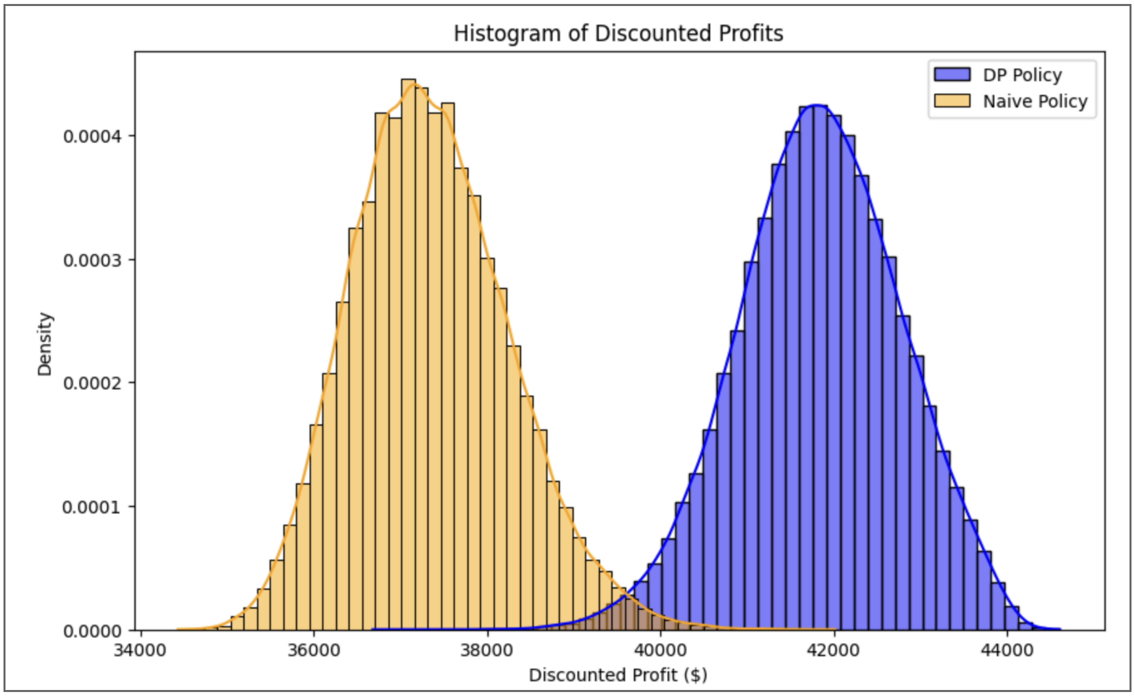
Seasonality Inclusion Policy

The Seasonality DP policy significantly improved profitability compared to the naive policy while effectively managing overbooking risks. Similar to the above comparison, the DP policy lowered the frequency and severity of overbooking incidents, reducing passenger dissatisfaction and associated costs substantially.

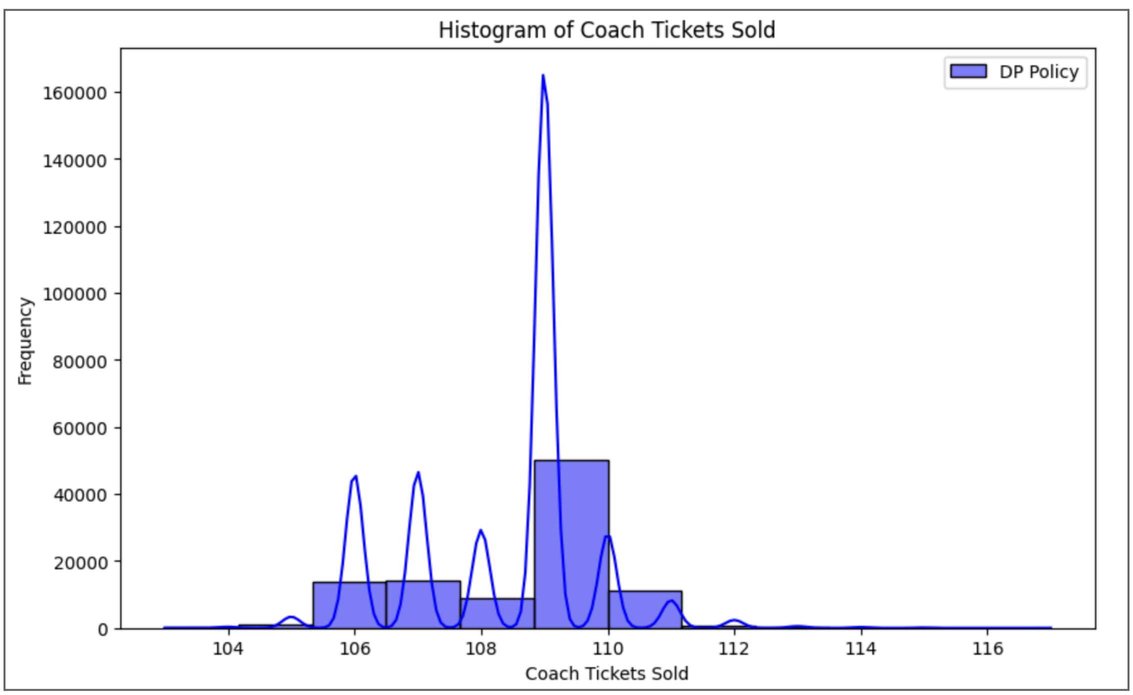
Metric	Naive Policy	Seasonality Policy
Expected Profit	$ 37,346.34	$ 41,833.27
Profit Volatility	$ 905.69	$ 932.13
Percentage Overbooked	100.00%	82.34%

Percentage Passengers Kicked Off	100.00%	70.99%
Average Overbooking Cost	$ 4,809.66	$ 852.63
Average Coach Tickets Sold	120.00	108.34
Average First-Class Tickets Sold	19.94	19.55

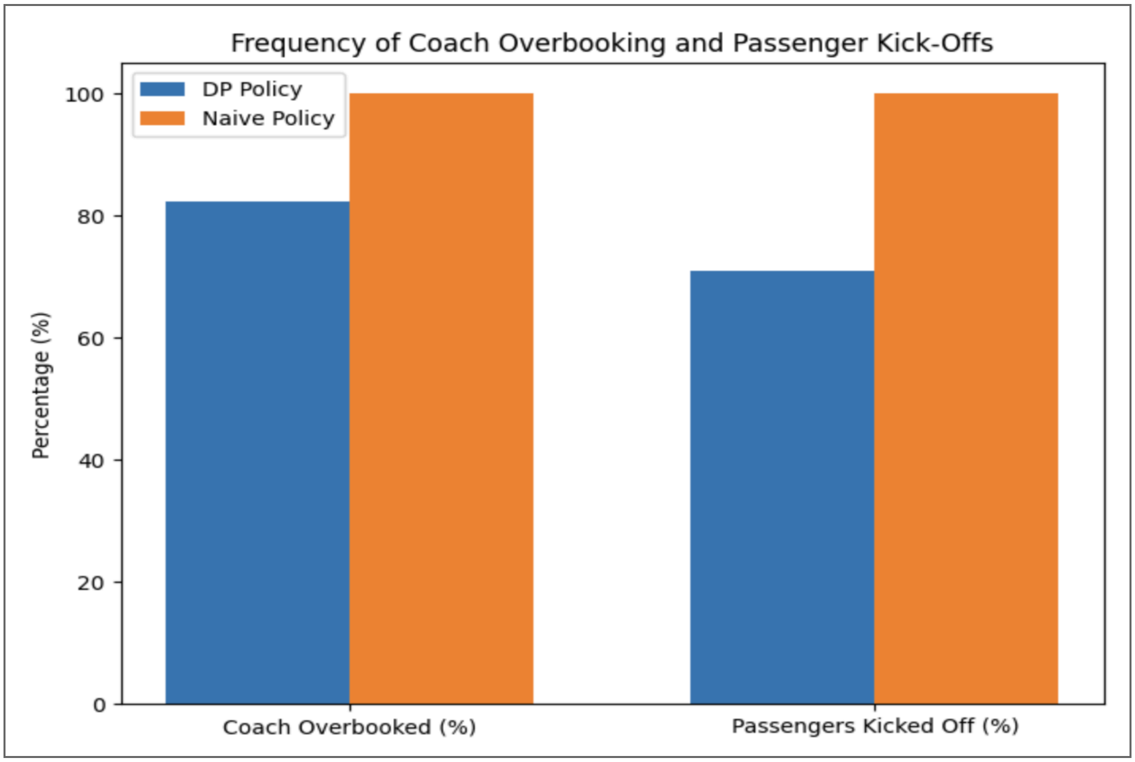




Histogram of Discounted Profits
DP Policy
Naive Policy
Density
0.0000
0.0001
0.0002
0.0003
0.0004
34000
36000
38000
40000
42000
44000
Discounted Profit ($)



Histogram of Coach Tickets Sold
DP Policy
Frequency
0
20000
40000
60000
80000
100000
120000
140000
160000
104
106
108
110
112
114
116
Coach Tickets Sold



Visual analysis from profit distribution plots and histograms confirmed that the DP strategy offered greater profit stability and more controlled sales management than the naive approach. These results emphasize the practical value of incorporating realistic demand seasonality into overbooking and pricing strategies.

Key Takeaways

The performance of the naive policy and four variations of DP-optimized strategies are as follows:

Metric	OB - 5	OB - 9	No Sale	Seasonality
Expected Profit	$38,148.03	$38381.68	$41,759.37	**$ 41,833.27**
Profit Volatility	$527.83	$833.59	$937.98	$ 932.13
% Overbooked	39.27%	90.26%	87.43%	82.34%
% Passengers Kicked Off	28.54%	80.85%	79.26%	70.99%

Avg Overbooking Cost	$194.28	$1,051.71	$1,020.36	$ 852.63
Avg Coach Tickets Sold	105.00	109	108.78	108.34
Avg First-Class Tickets Sold	19.87	19.67	19.75	19.55

Across all DP policies, we observe a consistent rightward shift in the distribution of discounted profits, indicating improved profitability. Even though DP strategies generally sell fewer tickets than the naive approach, they yield significantly higher profits due to more intelligent pricing, better handling of overbooking risk, and reduced passenger inconvenience.

Overbooking by 5 seats offers a safer low-risk, low-reward balance, with the lowest volatility and overbooking costs. Meanwhile, allowing overbooking by 9 seats with dynamic control achieves a slight uplift in expected profit while reducing overbooking costs by nearly 6% compared to the naive approach. The No-Sale and Seasonality DP policies further refine profitability, with Seasonality emerging as the most robust overall performer—balancing realistic demand fluctuations with consistent revenue gains. Both the No-Sale and Seasonality DP policies significantly outperformed the naive baseline, with profitability lifts of approximately 11.8% and 12.0%, respectively. Additionally, these policies reduced average overbooking costs dramatically—by 78.8% for the No-Sale policy and 82.3% for the Seasonality policy—compared to the naive policy.

Based on profitability uplift, reduction in overbooking risk, and realism in demand modeling, the **Seasonality DP Policy** is the most effective strategy to adopt. It consistently outperforms the naive approach and other DP variants across all key metrics, making it the ideal recommendation for implementation. T**he Seasonality DP Policy emerges as the optimal strategy**. It achieves the **highest profitability**, **lowest average overbooking cost among high-performing strategies**, and offers the **best balance between seat utilization and customer experience**.