Hotel booking Cancelation Prediction Model

by Muhammad Firza Alfajri
as part of learning journey in Data Scientist program at Purwadhika

Structure of the Presentation

- Case Overview
  - Business Problem
- Data preprocessing
  - Data Cleaning
  - Feature Engineering
- Analytics
  - Metrics
  - Modeling
- Conclusion
  - Cost-benefit analysis
  - Recommendation

Case Overview

Business Overview

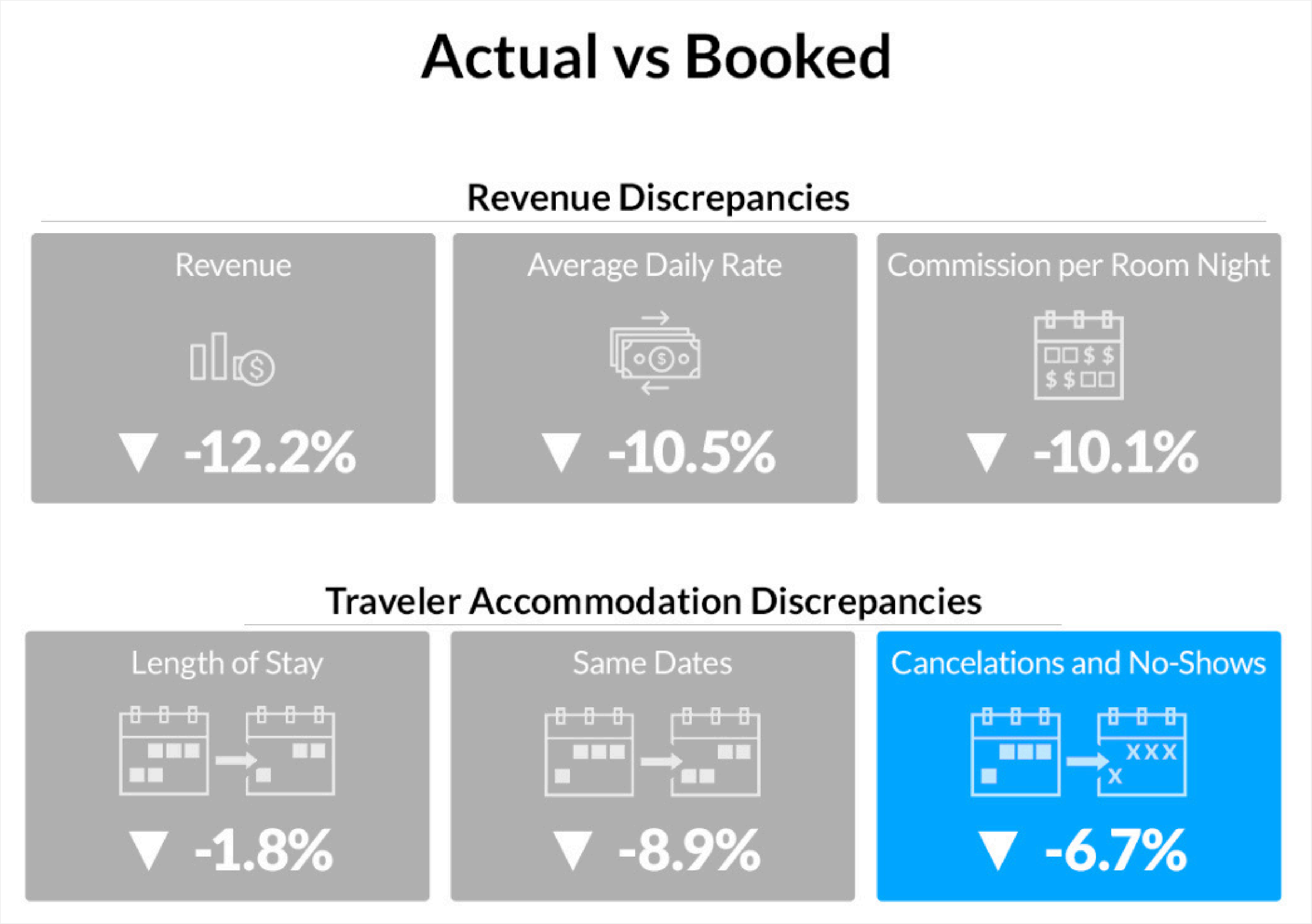
A **Mid-Sized Hotel** in Portugal

- Focused on development in **tourist** areas
- Utilizes **multiple reservation channels**

Stakeholders:

- Hotel manager
- Reservation management system
- Operation : reservation team

Business problem



https://skift.com/2024/07/23/summer-travels-hidden-hurdle-the-impacts-of-cancellations-and-no-shows/?utm_source=chatgpt.com

"A common discrepancy in hotel booking data comes from modifications after the initial reservation.
Onyx Insights data shows that roughly 10 percent of all bookings undergo modifications, representing a meaningful percentage of potential commissions.
Extrapolated out to the entire industry, Skift Research estimates that hotels paid more than $75 billion for indirect distribution costs in 2023. If 10 percent of all of those bookings see potential data discrepancies, that's a significant chunk of revenue that could be impacted."

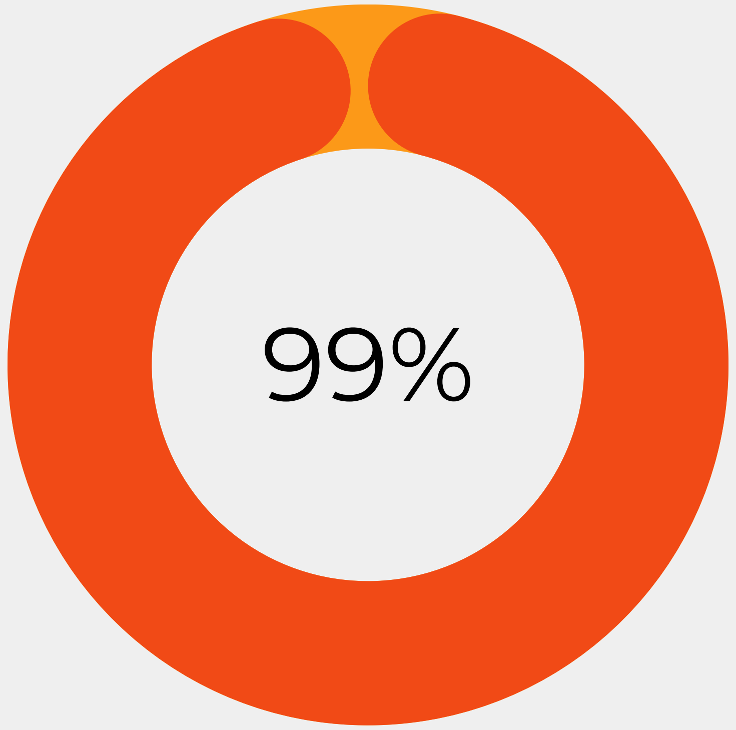
Problem Statement

“A high **cancellation** rate causes **revenue leakage** through **missed opportunity cost** and **wasted operational expense**.

This project will support preemptive effort to minimize cost and expense by developing a **classification machine learning** model to predict **booking cancellations** through recorded **booking behaviour**

Data preprocessing

Data Overview



83573 entries
351 missing values in country columns

Outliers are **included**

Initial Features	
country	days_in_waiting_list
market_segment	customer_type
previous_cancellations	reserved_room_type
booking_changes	required_car_parking_space
deposit_type	total_of_special_request
Target :	is_canceled

Data Cleaning

Columns	Row / Value	**method**	Justification
country	NaN	fillna	may help the ML model to recognize missing value in 'country'
country	Alpha 2 country code	replace with its Alpha 3 country code	will help in feature engineering later by mapping country into continent or subregion
Market_segment	Unidentified	drop	only 1 row contain this may identify test data

Feature engineering

Columns	feature generated	**description**
country	Subregion	change country unto its respective subregion
country	continent	change country unto its respective continent
Previous_cancelation	prev_cancelation_bin	bin into yes or no
Booking_changes	Booking_changes_bin	bin into yes or no
days_in_waiting_list	waitlist_bin	bin into yes or no
Total_special_request	request_bin	bin into yes or no

Feature Selection

Subregion	market_segment
deposit_type	prev_cancelation_bin
reserved_room_type	required_car_parking_spaces
Booking_changes_bin	waitlist_bin
total_of_special_requests	customer_type

Analytics: Metrics & Modeling

Metrics to be focused

0 : no cancelation 1: cancelation	Actual : True	Actual : False
Predicted: True	True Negative (+109$)	False Positive (-109$ for opportunity cost)
Predicted: False	False Negative (-109$ for opportunity cost and another -18$ for preparation = -127$)	True Positive (0$, but may be able to find/market the room)

According to : https://www.budgetyourtrip.com/hotels/portugal/lisbon-2267057

-

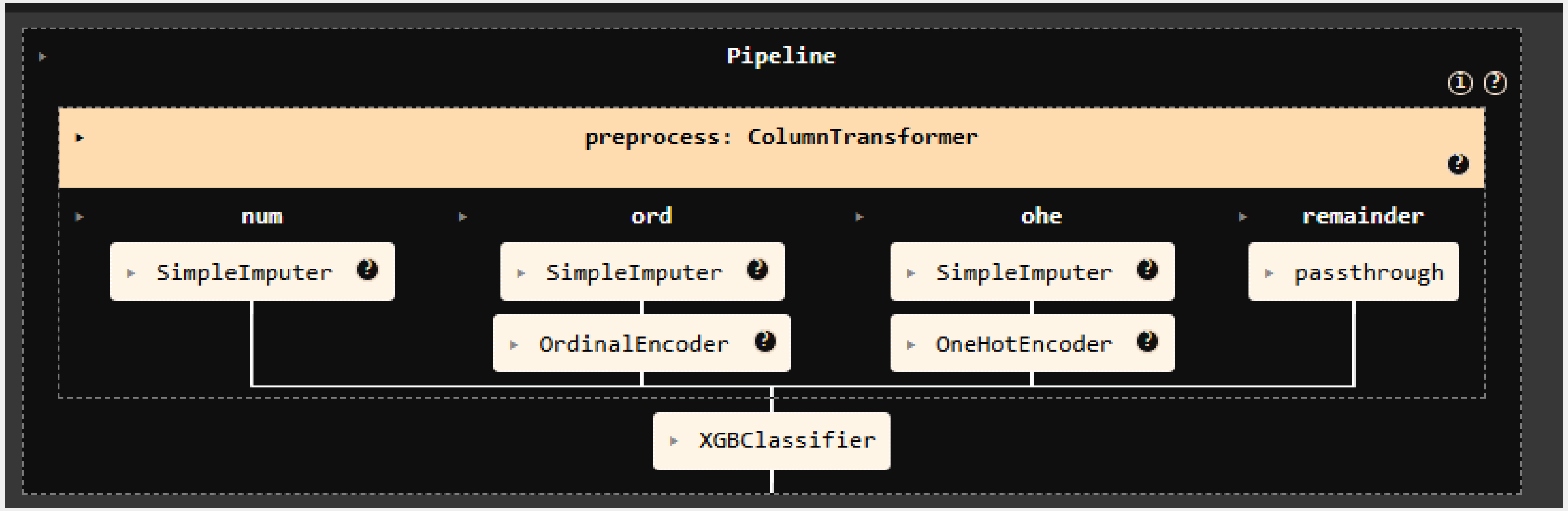
The average hotel price in Lisbon based on data from 1,691 hotels is an affordable $109 with high season averages around $204, and the median price is $96

roomkeeping cost need to bee calculated ar 18$ according to: https://www.hotelmanagement.net/housekeeping/what-true-cost-clean-guestroom?utm_source=chatgpt.com

Because FN are more costly, this model would choose **recall** as its main metrics

- Recall measures how many actual cancellations the model successfully identifies.
- the goal is to capture as many cancellations as possible to avoid losses from mistakenly assuming guests will show up.
- A model with high recall is less likely to make False Negative errors.

Pipeline



Data Preparation **Features (X)**: Selected from columns_3, representing relevant booking behavior. **Target (y)**: is_canceled – a binary label indicating whether a booking was canceled (1) or not (0). **Train-Test Split** **Training Set**: 70% of the data **Testing Set**: 30% of the data **Stratified Sampling**: Maintains the original cancellation rate in both training and testing datasets. **Random State = 42**: Ensures reproducibility of the split	**Feature Grouping:** **Numerical Features:** • total_of_special_requests, • required_car_parking_spaces **Ordinal Features:** • market_segment, deposit_type, customer_type, reserved_room_type, un_subregion **Binary / One-Hot Encoded Features:** • booking_changes_bin, prev_cancelation_bin, waitlist_bin	**Preprocessing Pipelines:** **Numerical Pipeline:** • Simple imputer (fill with 0) **Ordinal Pipeline:** • Imputer • OrdinalEncoder (preserve order, handle unknowns) **OHE Pipeline:** • Imputer • OneHotEncoder

Model

Model: XGBoost Classifier
A robust and scalable gradient boosting method for classification problems, especially with tabular data.

eval_metric='logloss'
- Log loss is used to optimize probabilistic predictions in binary classification.

use_label_encoder=False
- Avoids deprecation warnings and uses the updated scikit-learn API.

random_state=42
- Ensures reproducibility of results.

Parameter Tuning

1. n_estimators = [200]
- Number of boosting rounds (trees)
- fixed at 200 to reduce search complexity

2. max_depth = [11]
- Maximum depth of each tree
- Controls how detailed each decision rule can be

3. learning_rate = [0.1, 0.01]
- Step size at each boosting round
- Lower value = slower but more stable learning

4. scale_pos_weight = [ratio, 1.2 × ratio]
- Balances the importance of minority class (canceled bookings)
- Essential to handle class imbalance (e.g., far more non-cancelled than cancelled bookings)
- Ratio = (negative samples / positive samples)

Tuning Strategy

Grid Search with Cross Validation
- Cross-validated to ensure results generalize across folds (here: cv=3)

Refit Strategy: refit='recall'
- Model is re-trained using the hyperparameters that gave best recall
- Why Recall?
  - False negatives are costlier (missed cancellations)
  - Prioritizing recall helps minimize them

Model Evaluation

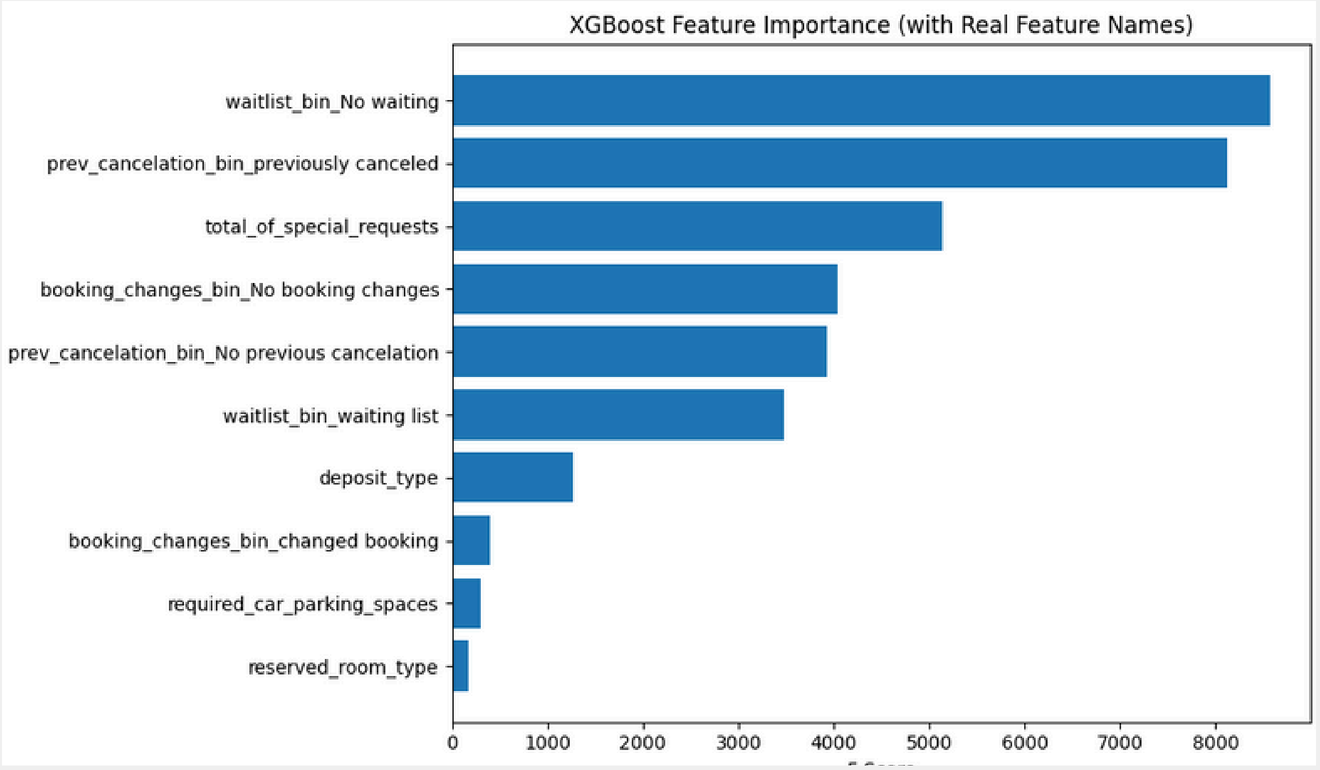
=== Test Classification Report (threshold = 0.5) ===

	precision	recall	f1-score	support
0	0.88	0.72	0.79	15839
1	0.64	0.83	0.72	9233
accuracy			0.76	25072
macro avg	0.76	0.78	0.76	25072
weighted avg	0.79	0.76	0.77	25072

Model Performance – Test Set (Threshold = 0.5)

- Class 0 – Not Canceled:
  - Precision 0.88 → Most predictions for non-canceled bookings are correct.
  - Recall 0.72 → It correctly captures 72% of actual non-canceled bookings.
- Class 1 – Canceled Bookings (our focus):
  - Recall 0.83 → Model successfully identifies 83% of actual cancellations, which is crucial to prevent losses.
  - Precision 0.64 → Among those predicted as cancellations, 64% are correct.
- Overall Accuracy: 76%
  - But more importantly, macro and weighted F1-scores are around 76–77%, reflecting good balance across both classes.

Conclusion: The model is tuned for high recall on cancellations, making it effective in flagging most risky bookings for follow-up actions.



Feature Importance

- Most Important Features:
  - waitlist_bin_No waiting and prev_cancelation_bin_previously canceled are the strongest signals.
  - total_of_special_requests ranks third — guests making fewer special requests tend to cancel more, indicating lower commitment.
- Other Key Factors:
  - Lack of booking changes (booking_changes_bin_No booking changes) and no prior cancellations (prev_cancelation_bin_No previous cancelation) also play a role.
  - Interestingly, even deposit type and room type contribute, though to a lesser extent.

Takeaway: The model focuses on behavioral indicators (waitlists, past cancellations, booking changes) — which are valuable for early risk detection and proactive engagement.

Conclusion

Cost-benefit Analysis

0 : no cancelation 1: cancelation	**Predicted: false**	**Predicted: true**
Actual : false	True negative: 11420	False Positive 4419
Actual : true	False Negative 1531	True positve 7702

Revenue gain:
TN: $109 * 11420 = 1,244,780
Loss from:
FN: $127 * 1531 = $194,437
FP: $109 * 4419 =$481,671
total cost : $ 676,108
Net Impact : $568.672

with no model, assump all cancelation are not predicted:
FP + TP = 1531 + 7702 = 9,233 * $127 = $1.172.591
benefit gained:
$1,172,591 - $676,108 = $496.483

Recommendation

- Model performs well (recall-focused → detects cancellations accurately)
- Ready for integration into reservation system
- Threshold tuning = customize based on season or occupancy target
- Set alerts for high-risk bookings → proactive action

End

Thank you

Do you have any questions?