CAPSTONE MODUL 3

HOTEL BOOKING CANCELLATIONS PREDICTION

FIKA NABILA
JCDS 2804-019

BUSSINESS PROBLEM UNDERSTANDING

Latar Belakang

Hotel di Portugal ingin mengurangi kerugian akibat pembatalan pemesanan kamar. Diperlukan model prediksi untuk mengidentifikasi pelanggan yang kemungkinan besar akan membatalkan pemesanannya.

Masalah

Bagaimana memprediksi apakah seorang pelanggan akan membatalkan pemesanan kamar berdasarkan informasi saat reservasi?

Tujuan

- Membangun model klasifikasi pembatalan pemesanan
- Mengidentifikasi fitur yang memengaruhi pembatalan
- Memberikan rekomendasi untuk mengurangi cancelation rate dan optimalkan pendapatan

PENDEKATAN ANALITIK

- Eksplorasi dan visualisasi fitur-fitur penting
- Penanganan imbalance dengan SMOTE
- Model Machine Learning (LightGBM + tuning)
- F1-Score (kelas 1) sebagai metrik utama dan ROC AUC sebagai metrik tambahan

Jenis Error	Nama Statistik	Dampak
Type 1 Error	False Positive	Potensi hilangnya pendapatan dan kepercayaan pelanggan
Type 2 Error	False Negative	Kamar kosong mendadak, rugi operasional

country	Negara pelanggan
market_segment	Segmen pasar (online/offline/corporate)
previous_cancellations	Jumlah pembatalan sebelumnya
booking_changes	Jumlah perubahan pemesanan
deposit_type	Jenis deposit
days_in_waiting_list	Hari dalam daftar tunggu
customer_type	Jenis pelanggan
reserved_room_type	Tipe kamar dipesan
required_car_parking_space	Jumlah tempat parkir diminta
total_of_special_request	Jumlah permintaan khusus
is_canceled	Status pembatalan (1=ya, 0=tidak)

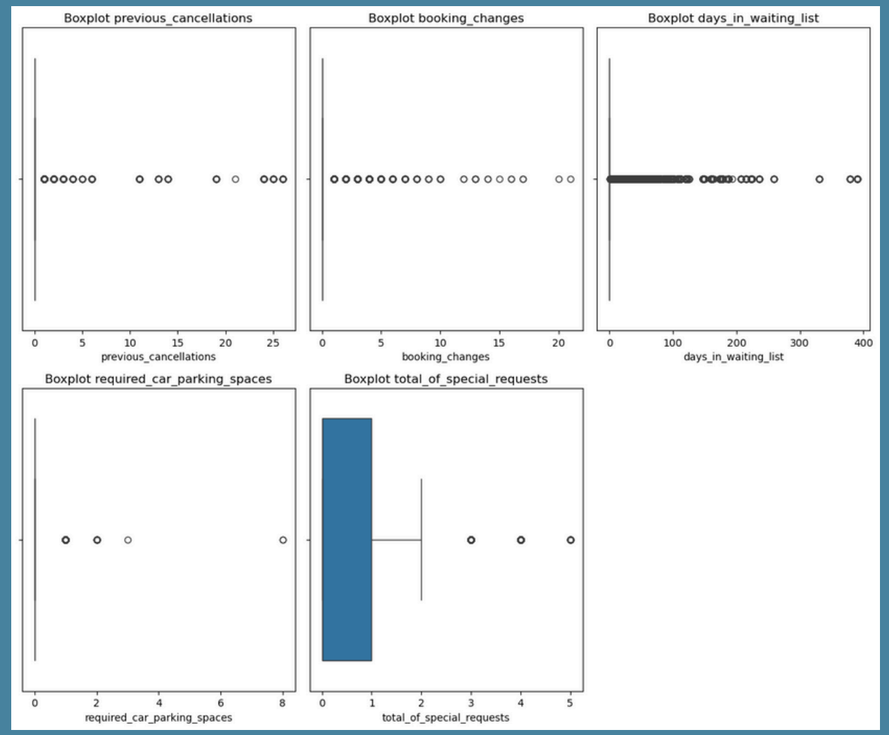
DATA UNDERSTANDING

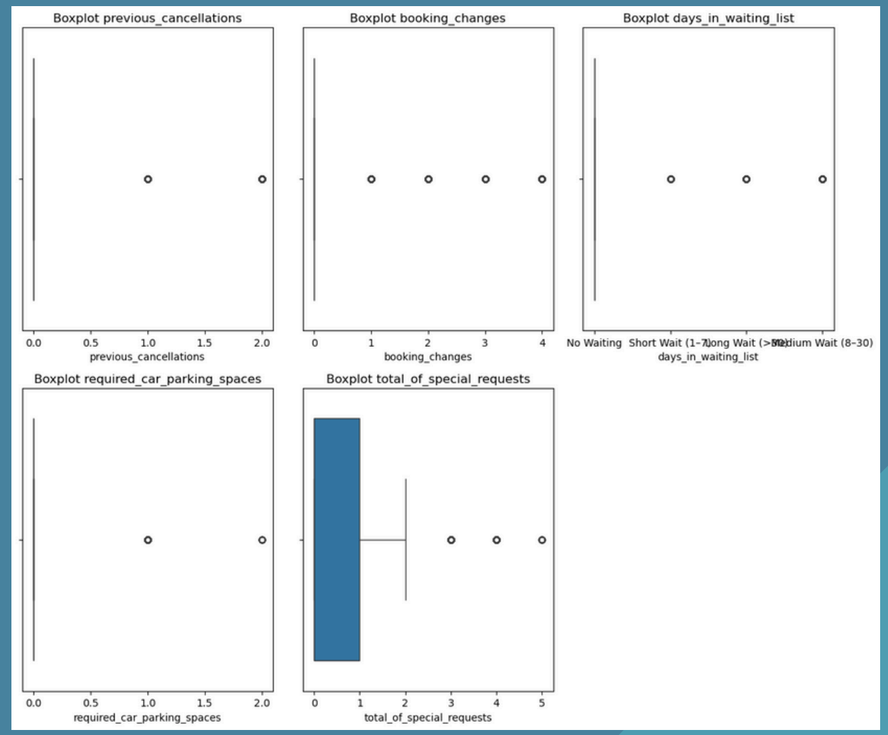
Dataset ini berisi informasi pemesanan kamar hotel di Portugal dan mencakup berbagai karakteristik pelanggan serta detail reservasi mereka.
Seluruh informasi yang dapat mengidentifikasi pelanggan secara pribadi telah dihapus demi menjaga privasi.

```
is_canceled
0    63.172316
1    36.827684
Name: proportion, dtype: float64
```

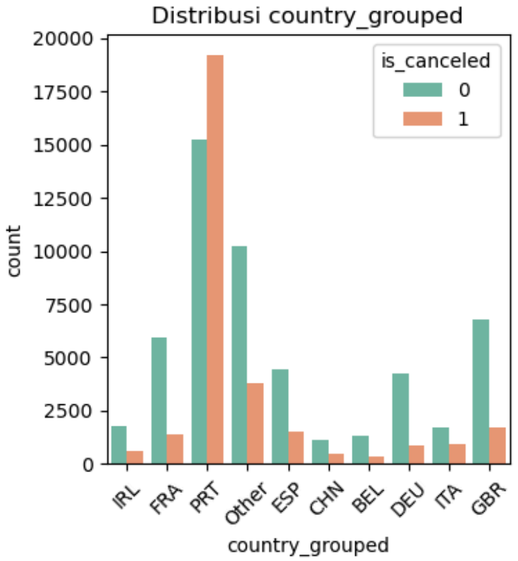
DATA CLEANING

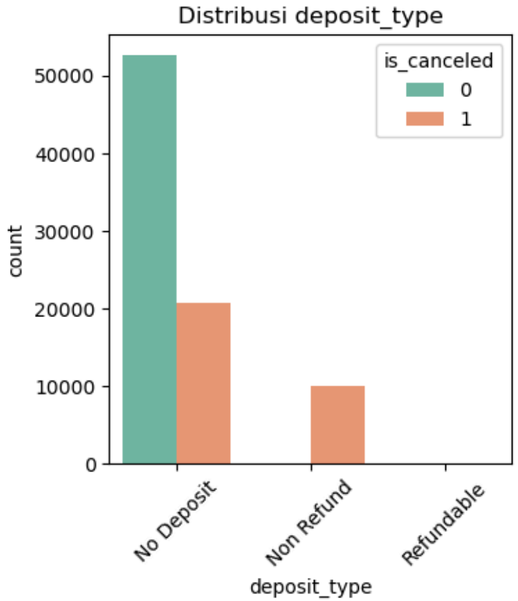
- Duplikat dibiarkan (87% data)
- Imputing missing value country dengan mode
- Standarisasi kode negara sesuai ISO 3166-1 alpha-3
- Binning kolom dengan frekuensi rendah
- Kendalikan outlier numerik dengan binning dan clipping
- Binning kolom days_in_waiting_list menjadi 4 kategori

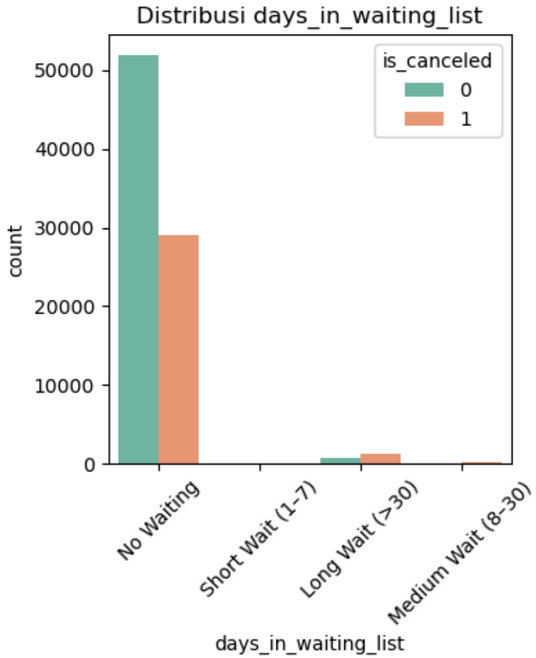


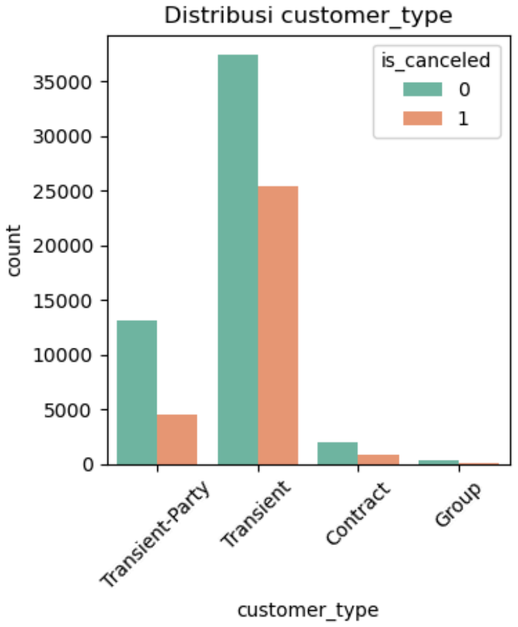


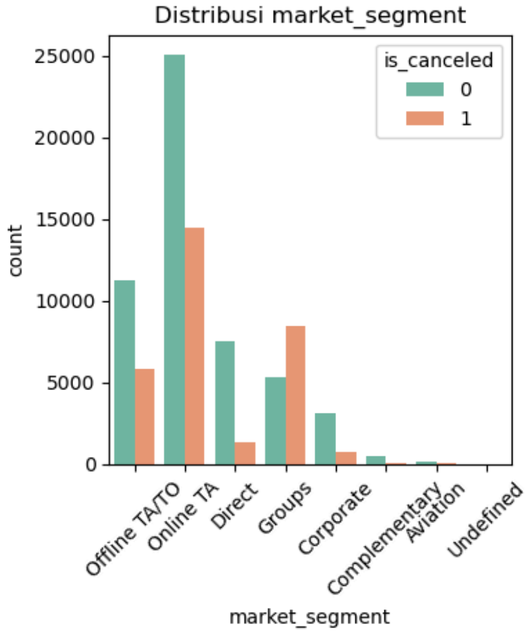
Distribusi Fitur Kategorikal Berdasarkan Status Pembatalan

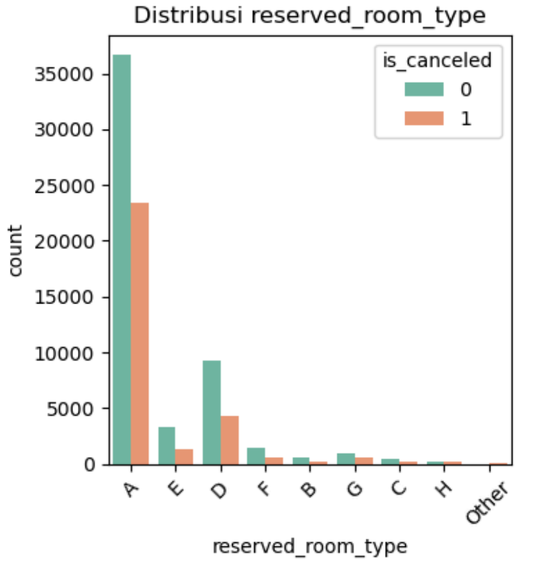




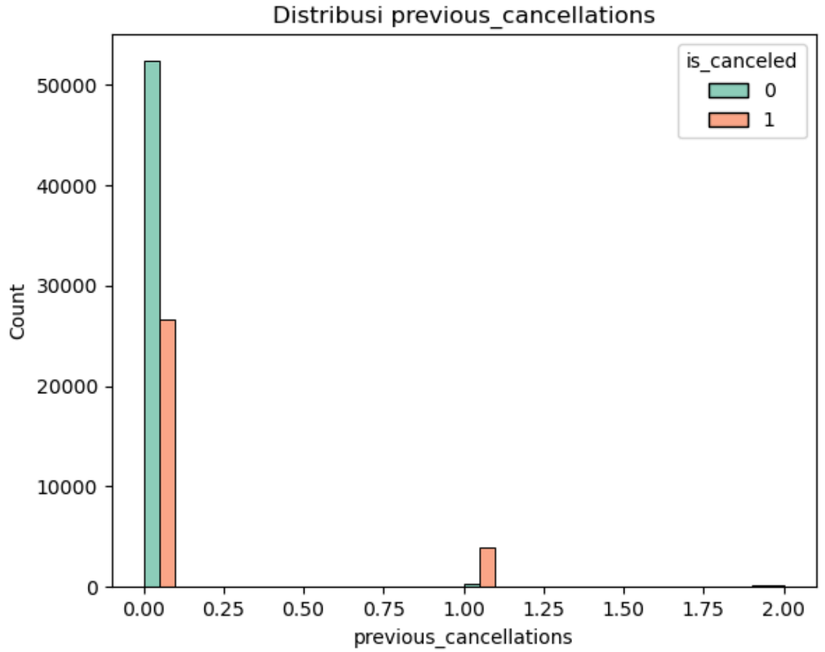


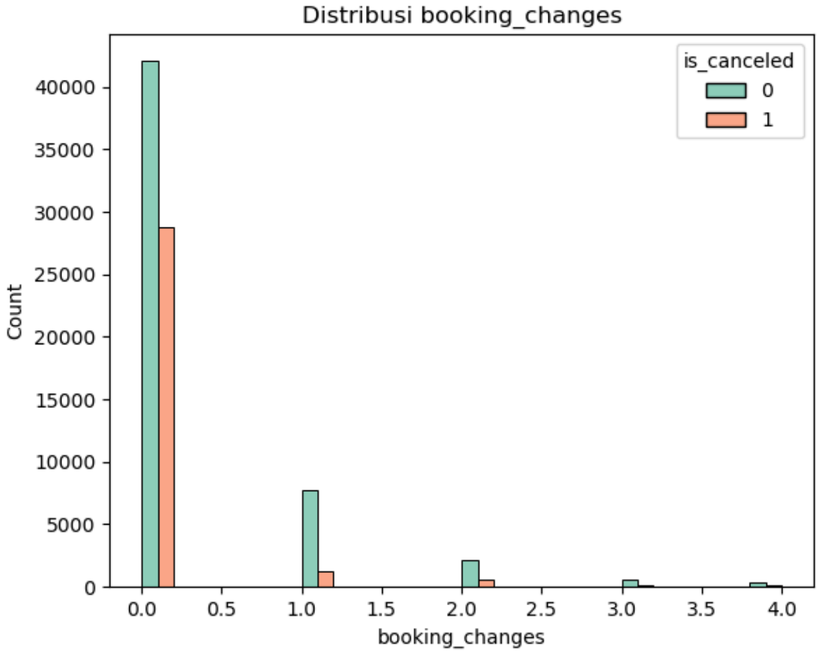


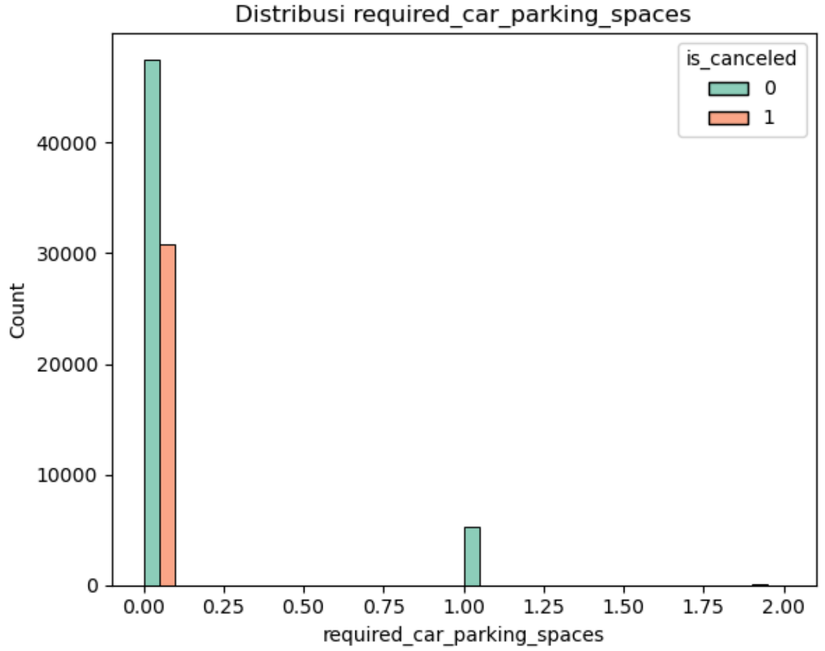


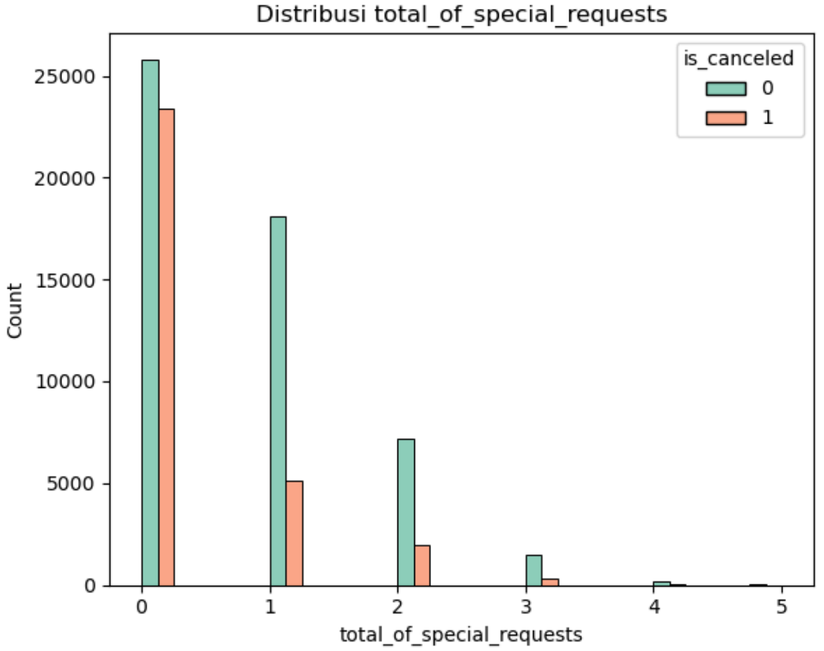


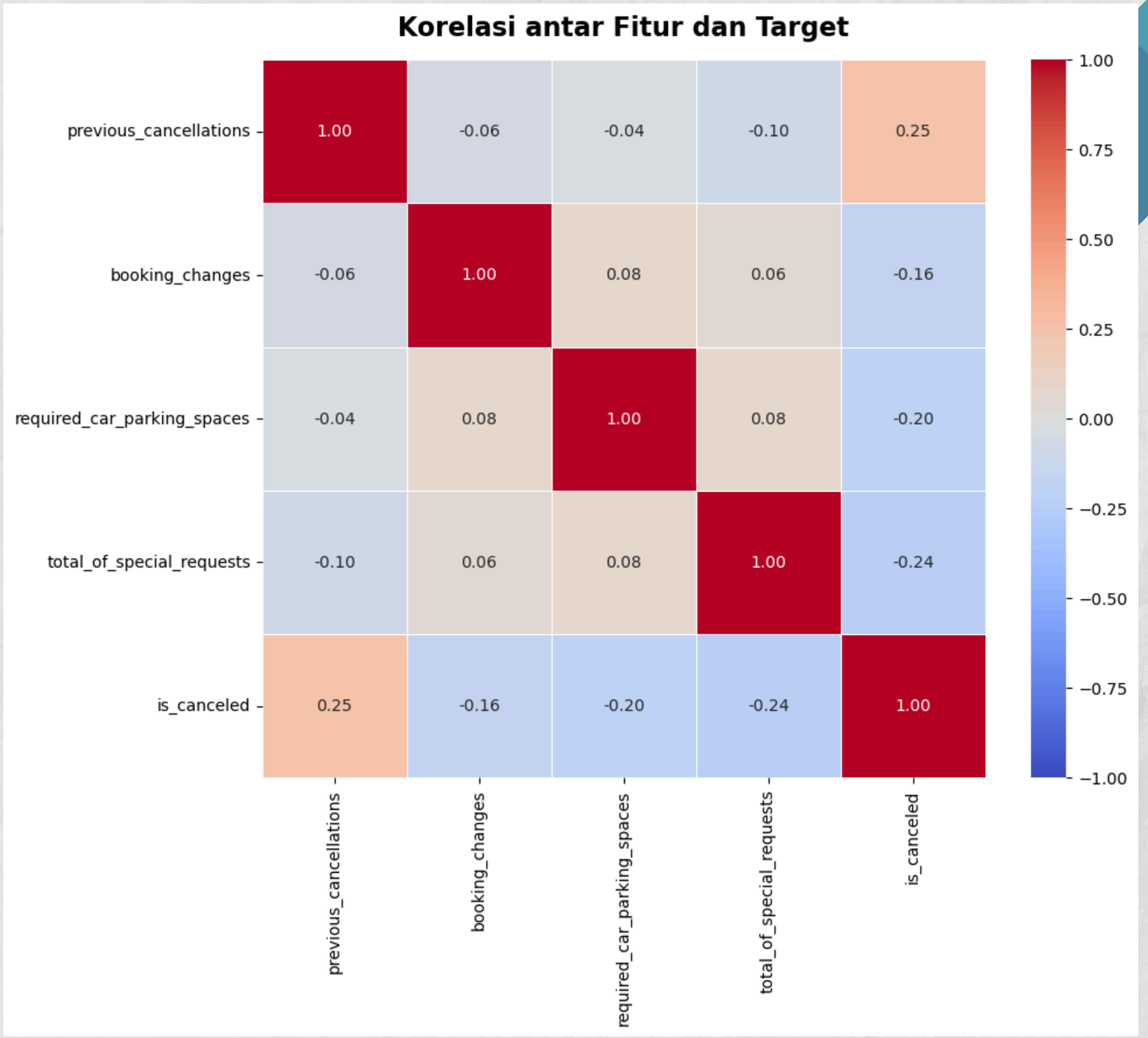
Distribusi Fitur Numerik Berdasarkan Status Pembatalan










Korelasi antar Fitur dan Target
previous_cancellations
booking_changes
required_car_parking_spaces
total_of_special_requests
is_canceled
1.00 -0.06 -0.04 -0.10 0.25
-0.06 1.00 0.08 0.06 -0.16
-0.04 0.08 1.00 0.08 -0.20
-0.10 0.06 0.08 1.00 -0.24
0.25 -0.16 -0.20 -0.24 1.00
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00

DATA PREPARATION

```python
# Fitur & Target
X = df.drop('is_canceled', axis=1)
y = df['is_canceled']
```

```python
# Train-test split
X_train, X_test, y_train, y_test = train_test_split(X, y, stratify=y, test_size=0.2, random_state=2021)
```

```python
# Preprocessing
frequency_cols = ['country']
ordinal_cols = ['days_in_waiting_list']
onehot_cols = ['market_segment', 'deposit_type', 'customer_type', 'reserved_room_type']
numeric_cols = ['previous_cancellations', 'booking_changes', 'required_car_parking_spaces', 'total_of_special_requests']

# Pipeline fitur kategorikal
def frequency_encode_country(X):
    series = pd.Series(X.ravel())
    freq = series.value_counts()
    return series.map(freq).values.reshape(-1, 1)

frequency_pipeline = Pipeline([
    ('imputer', SimpleImputer(strategy='constant', fill_value='most_frequent')),
    ('frequency', FunctionTransformer(frequency_encode_country, validate=False))
])

ordinal_pipeline = Pipeline([
    ('ordinal', OrdinalEncoder(categories=[['No Waiting', 'Short Wait (1–7)', 'Medium Wait (8–30)', 'Long Wait (>30)']]))
])

onehot_pipeline = Pipeline([
    ('onehot', OneHotEncoder(handle_unknown='ignore'))
])

# Pipeline fitur numerik
numeric_pipeline = Pipeline([
    ('imputer', SimpleImputer(strategy='mean')),
    ('scaler', StandardScaler())
])

# Pipeline preprocessing
preprocessor = ColumnTransformer([
    ('frequency', frequency_pipeline, frequency_cols),
    ('ordinal', ordinal_pipeline, ordinal_cols),
    ('onehot', onehot_pipeline, onehot_cols),
    ('numeric', numeric_pipeline, numeric_cols)
])
```

MODELLING

Model Benchmarking : K-Fold

Model	# F1 Mean (Class 1)	# F1 Std	# ROC AUC Mean	# ROC AUC Std
LightGBM	0.683	0.006	0.837	0.004
XGBoost	0.679	0.006	0.836	0.004
Random Forest	0.66	0.004	0.816	0.002
Decision Tree	0.64	0.005	0.792	0.006
Logistic Regression	0.595	0.009	0.843	0.002
KNN	0.532	0.011	0.693	0.01

Model Resampling with SMOTE and SMOTEENN

	Model	# F1 Mean ⋯	# F1 Mean (SMOTE)	# F1 Mean (SMOTEENN)	# AUC Mean	# AUC Mean (SMOTE)	# AUC Mean (SMOTEENN)
5	LightGBM	0.683	0.689	0.667	0.837	0.84	0.815
4	XGBoost	0.679	0.687	0.663	0.836	0.841	0.819
3	Random Forest	0.66	0.672	0.653	0.816	0.816	0.762
2	Decision Tree	0.64	0.654	0.643	0.792	0.791	0.726
0	Logistic Regression	0.595	0.693	0.688	0.843	0.843	0.833
1	KNN	0.532	0.534	0.377	0.693	0.689	0.596

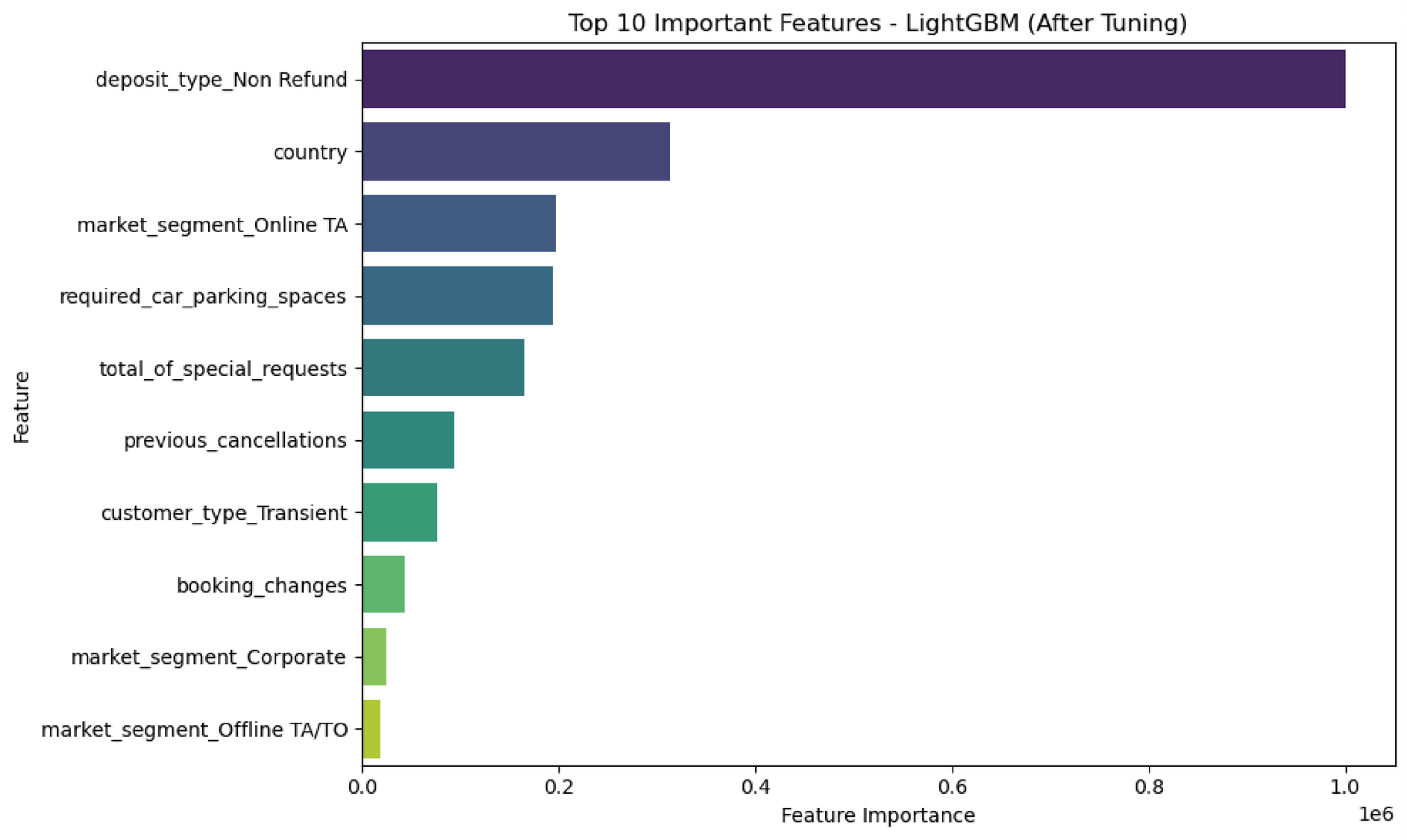
Hyperparameter Tuning

Best Parameters: {'model__num_leaves': 31, 'model__n_estimators': 50, 'model__max_depth': 5, 'model__learning_rate': 0.1}

	Model	# F1 Mean	# F1 Std	# AUC Mean	# AUC Std
0	LightGBM SMOTE (Before Tuning)	0.689	0.002	0.84	0.004
1	LightGBM SMOTE (After Tuning)	0.693	0.002	0.834	0.002

Saving the Model with Pickle

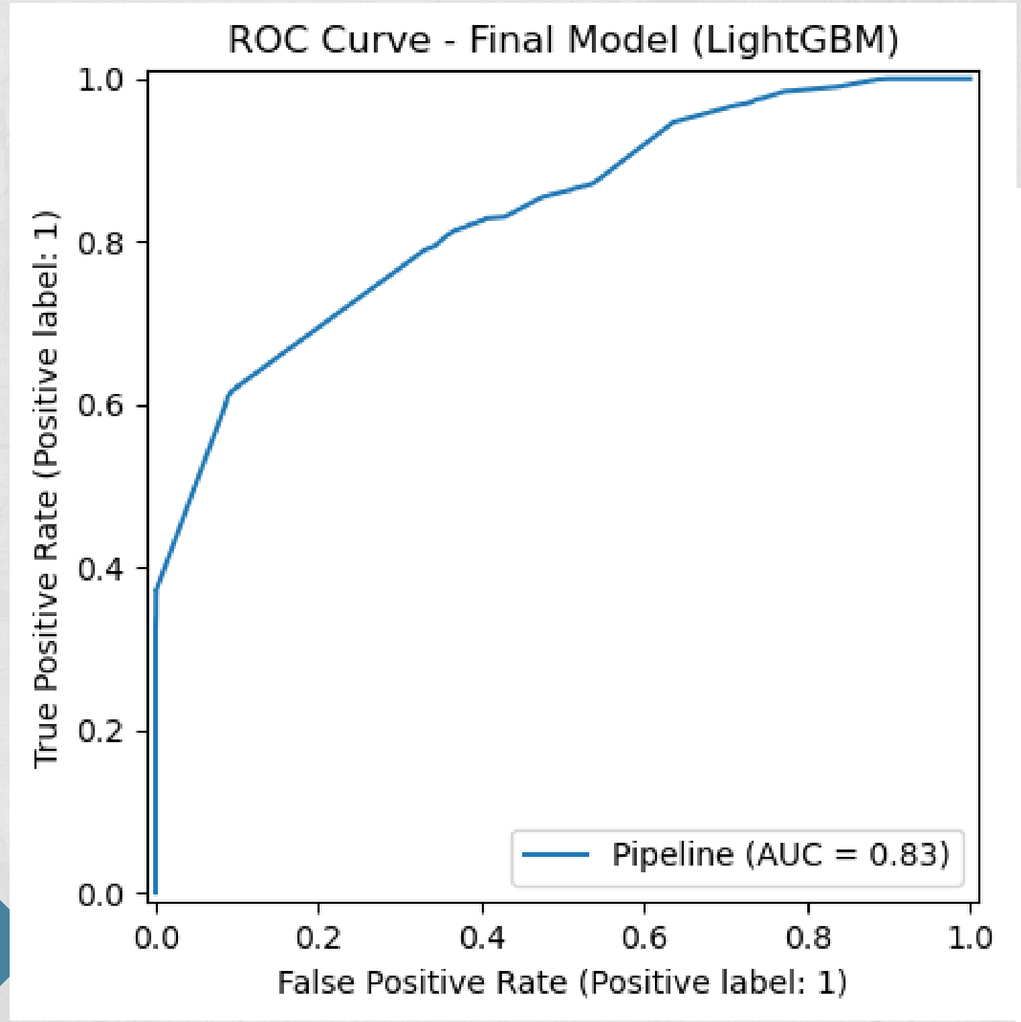
Feature Importance

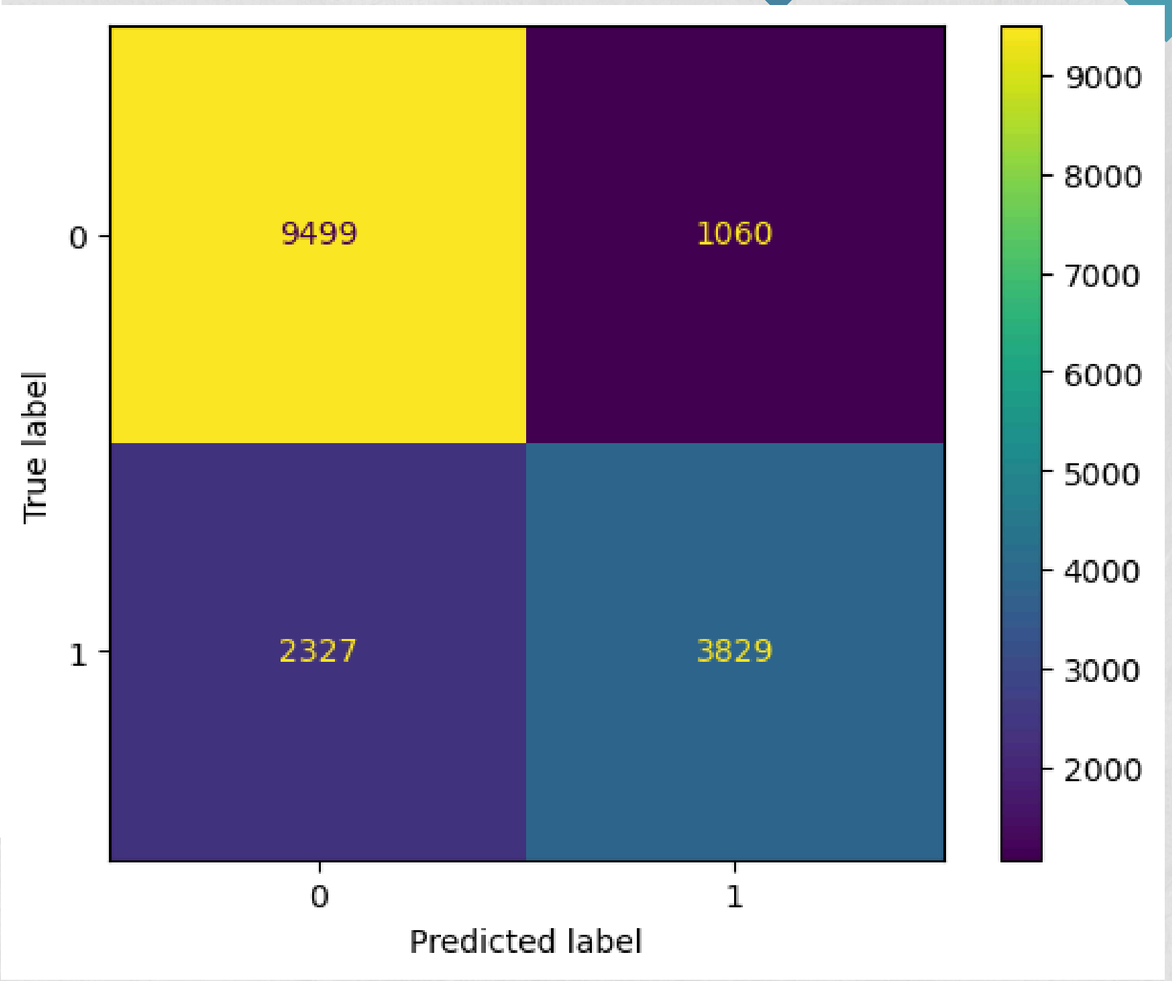


EVALUATION IN DATA TEST

Classification Report:

	precision	recall	f1-score	support
0	0.80	0.90	0.85	10559
1	0.78	0.62	0.69	6156
accuracy			0.80	16715
macro avg	0.79	0.76	0.77	16715
weighted avg	0.80	0.80	0.79	16715


ROC Curve - Final Model (LightGBM)
True Positive Rate (Positive label: 1)
False Positive Rate (Positive label: 1)
Pipeline (AUC = 0.83)
0.0
0.2
0.4
0.6
0.8
1.0


True label
Predicted label
9499
1060
2327
3829
0
1
9000
8000
7000
6000
5000
4000
3000
2000

SIMULASI DAMPAK FINANSIAL MODEL PREDIKSI PEMBATALAN HOTEL

Asumsi Dasar

- Total reservasi: 200 tamu (100 batal, 100 tidak)
- Kerugian per pembatalan: $185
- Model prediksi: Recall = 62%, Precision = 78%

Skenario	Jumlah	Total
Tanpa model (100 batal)	100	$18,500 kerugian
Dengan model – batal dicegah	62	$11,470 penghematan
Dengan model – false positive	18	$3,330 kerugian baru
Net Saving (Bersih)	-	$8,140

RECOMMENDATION

Data Recommendation

Country :

- Imputasi missing value & standarisasi kode.
- Kelompokkan berdasarkan wilayah geografis.

Outlier :

- Khususnya kolom days_in_waiting_list – analisis atau lakukan binning.

Fitur Temporal :

- Manfaatkan informasi tanggal (booking/check-in) sebagai fitur prediktif baru.

Model Recommendation

- Cost-sensitive learning → penalti lebih besar untuk FN.
- Feature engineering → interaction (e.g., deposit × segment).
- Ensemble stacking → kombinasi model (LightGBM + Logistic + CatBoost).
- SHAP → untuk interpretasi dan seleksi fitur.
- Hyperparameter tuning lanjutan → Optuna / Bayesian Optimization.

THANK YOU