Hotel Booking Cancellation Prediction

FATMA SABRY

READING THE DATA

2	INN00003	2	1	1	3	Meal Plan 1	0	Room_Type 1	1	Online	0	0	0	50.00	0	2/28/2018	Canceled
3	INN00004	1	0	0	2	Meal Plan 1	0	Room_Type 1	211	Online	0	0	0	100.00	1	5/20/2017	Canceled

```
import pandas as pd
df = pd.read_csv('first inten project.csv')
df.head()
```

This table shows a snapshot of the hotel booking dataset after loading it using pandas.
Each row represents a unique booking, including information like number of adults, children, meal plan, room type, booking lead time, and the final booking status (Canceled or Not_Canceled).

These sample rows help us understand the structure and contents of the dataset before we begin preprocessing.

EXPLORATORY DATA ANALYSIS

	Booking_ID	number of adults	number of children	number of weekend nights	\
0	INN00001	1	1	2	
1	INN00002	1	0	1	
2	INN00003	2	1	1	
3	INN00004	1	0	0	
4	INN00005	1	0	1	

	number of week nights	type of meal	car parking space	room type	\
0	5	Meal Plan 1	0	Room_Type 1	
1	3	Not Selected	0	Room_Type 1	
2	3	Meal Plan 1	0	Room_Type 1	
3	2	Meal Plan 1	0	Room_Type 1	
4	2	Not Selected	0	Room_Type 1	

I loaded and explored the dataset containing 36,285 rows and 17 columns.
Each row represents one hotel booking with features like number of guests, nights stayed, price, and more.
No missing values were found — all columns are complete.
Data includes:
10 numerical columns (e.g., number of adults, children, lead time)
6 categorical columns (e.g., room type, meal plan)
1 float column (average price

MISSING VALUES CHECK

```
print(df.isnull().sum())
```

```
Booking_ID                      0
number of adults                0
number of children              0
number of weekend nights        0
number of week nights           0
type of meal                    0
car parking space               0
room type                       0
lead time                       0
market segment type             0
repeated                        0
P-C                             0
P-not-C                         0
average price                   0
special requests                0
date of reservation             0
booking status                  0
dtype: int64
```

After running df.isnull().sum(), we confirmed that there are no missing values in any of the 17 columns.
→ This means the dataset is complete and no imputation is required

DATA TYPES SUMMARY

```
print(df.dtypes)
```

```
Booking_ID                    object
number of adults               int64
number of children             int64
number of weekend nights       int64
number of week nights          int64
type of meal                  object
car parking space              int64
room type                     object
lead time                      int64
market segment type           object
repeated                       int64
P-C                            int64
P-not-C                        int64
average price                float64
special requests               int64
date of reservation           object
booking status                object
dtype: object
```

- Total columns: 17
- Numerical Columns (int/float): 11
- e.g. number of adults, lead time, average price
- Categorical Columns (object): 6
- e.g. type of meal, room type, booking status

massa, eu convallis est.

Float Columns: 1 → average price

CATEGORICAL COLUMNS

```
cat_cols = df.select_dtypes(include=['object']).columns
for col in cat_cols:
    print(f"Column: {col}")

print(df[col].str.strip().unique())
```

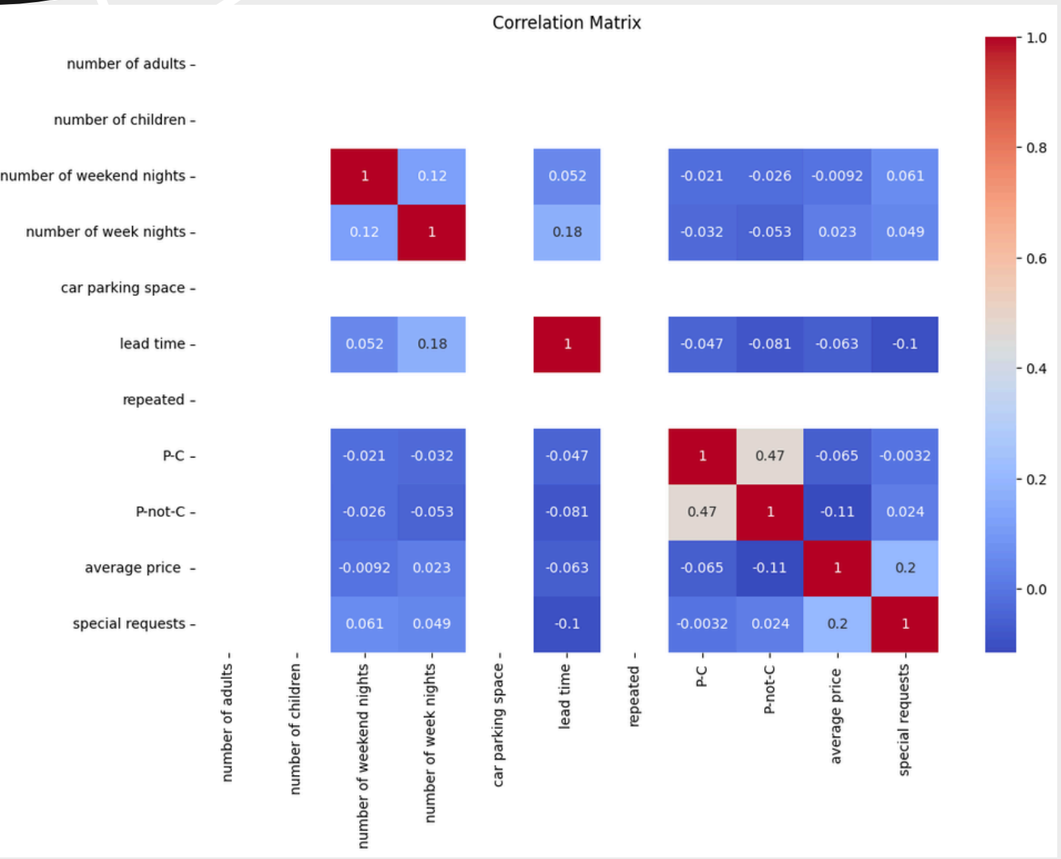
Booking_ID → 36,285 unique values (each booking has a unique ID)
type of meal → 4 categories: Meal Plan 1, Meal Plan 2, Meal Plan 3, Not Selected
room type → 7 different room types: Room_Type 1 to Room_Type 7
market segment type → 5 sources: Offline, Online, Corporate, Aviation, Complementary
date of reservation → stored as string, will be converted to datetime later
booking status → target variable with two classes: Not_Canceled and Canceled
All categorical values are clean and contain no extra whitespace

OUTLIER DETECTION AND TREATMENT

Outliers in average price : 1696
Outliers in car parking space : 1124
Outliers in lead time : 1332
Outliers in number of adults : 10175
Outliers in number of children : 2702
Outliers in number of week nights : 324
Outliers in number of weekend nights : 21
Outliers in repeated : 930
Outliers in special requests : 762

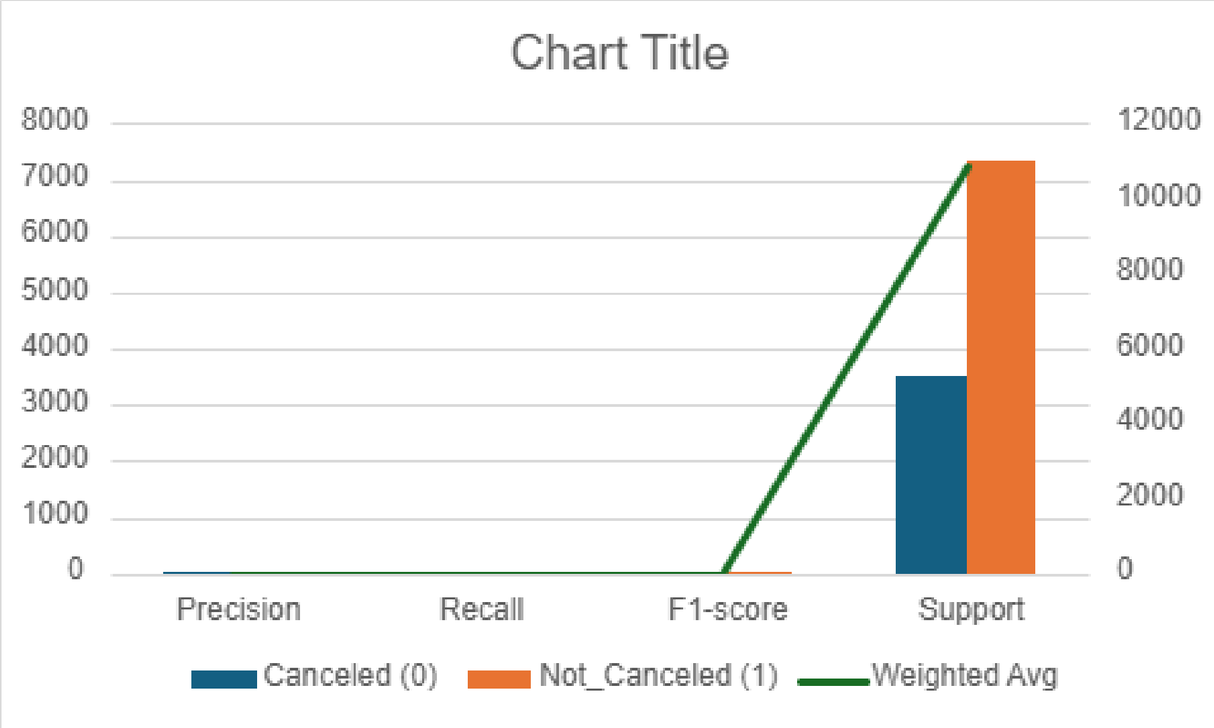
```
Outliers in average price : 1696
Outliers in car parking space: 1124
Outliers in lead time: 1332
Outliers in number of adults: 10175
Outliers in number of children: 2702
Outliers in number of week nights: 324
Outliers in number of weekend nights: 21
Outliers in repeated: 930
Outliers in special requests: 762
```

CORRELATION ANALYSIS & COLUMN PRUNING



The heat-map reveals relationships among numeric features and highlights any multicollinearity.
Booking_ID, P-C and P-not-C are identifiers or aggregate counts that do not add predictive power, so they were dropped to simplify the dataset.
Remaining features provide a cleaner input for feature engineering and modeling.

MODEL 1: LOGISTIC REGRESSION RESULTS



The model performs well in identifying non-canceled bookings, with a high recall of 89%

It struggles more with canceled bookings, achieving only 61% recall

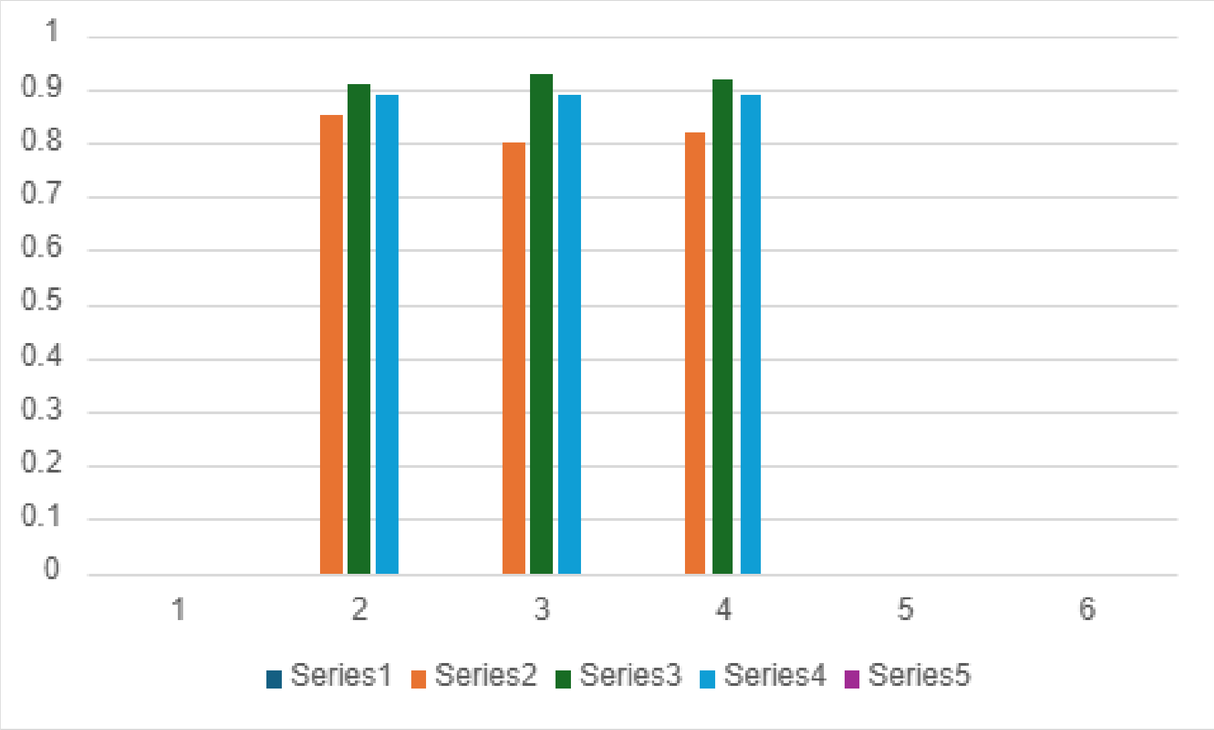
The overall accuracy of the model is approximately 80%

Weighted F1-score is 0.79, indicating balanced performance across classes

A warning appeared suggesting increasing max_iter for better convergence

RANDOM FOREST CLASSIFIER RESULTS

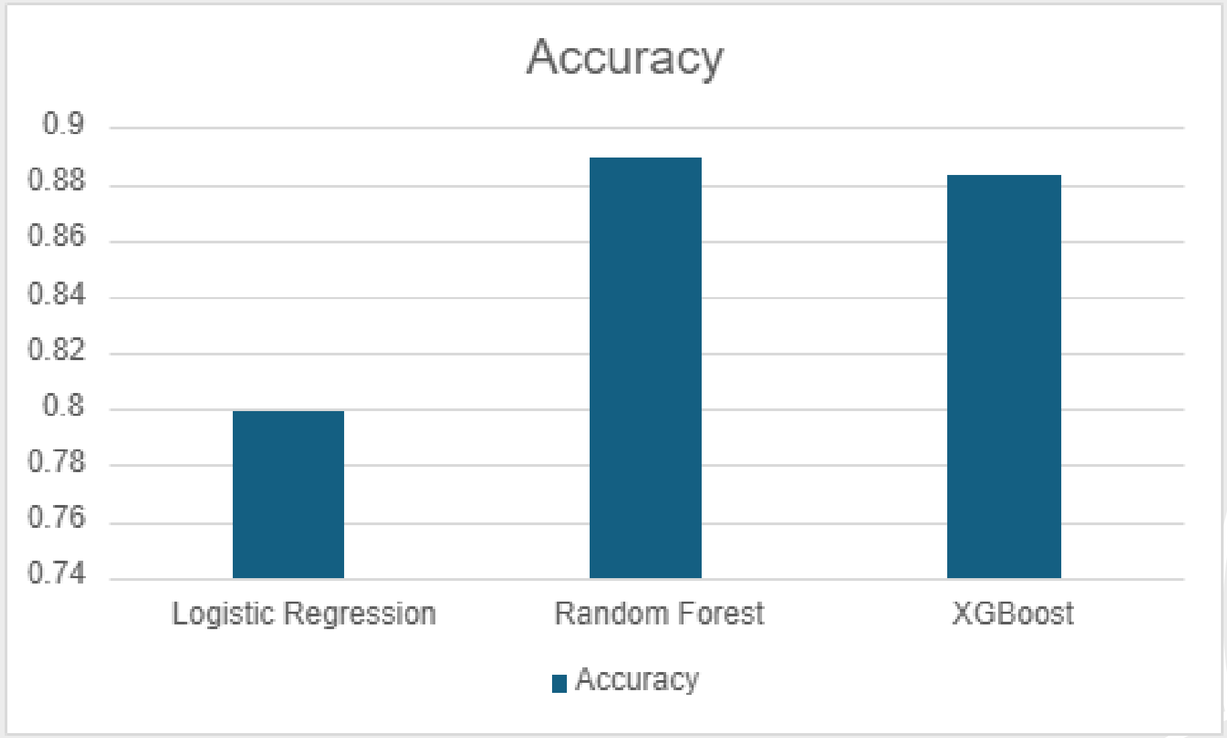
4	Class	Precision	Recall	F1-score	Support
5	Canceled (0)	0.85	0.8	0.82	3517
6	Not_Canceled (1)	0.91	0.93	0.92	7358
7	Weighted average	0.89	0.89	0.89	10875



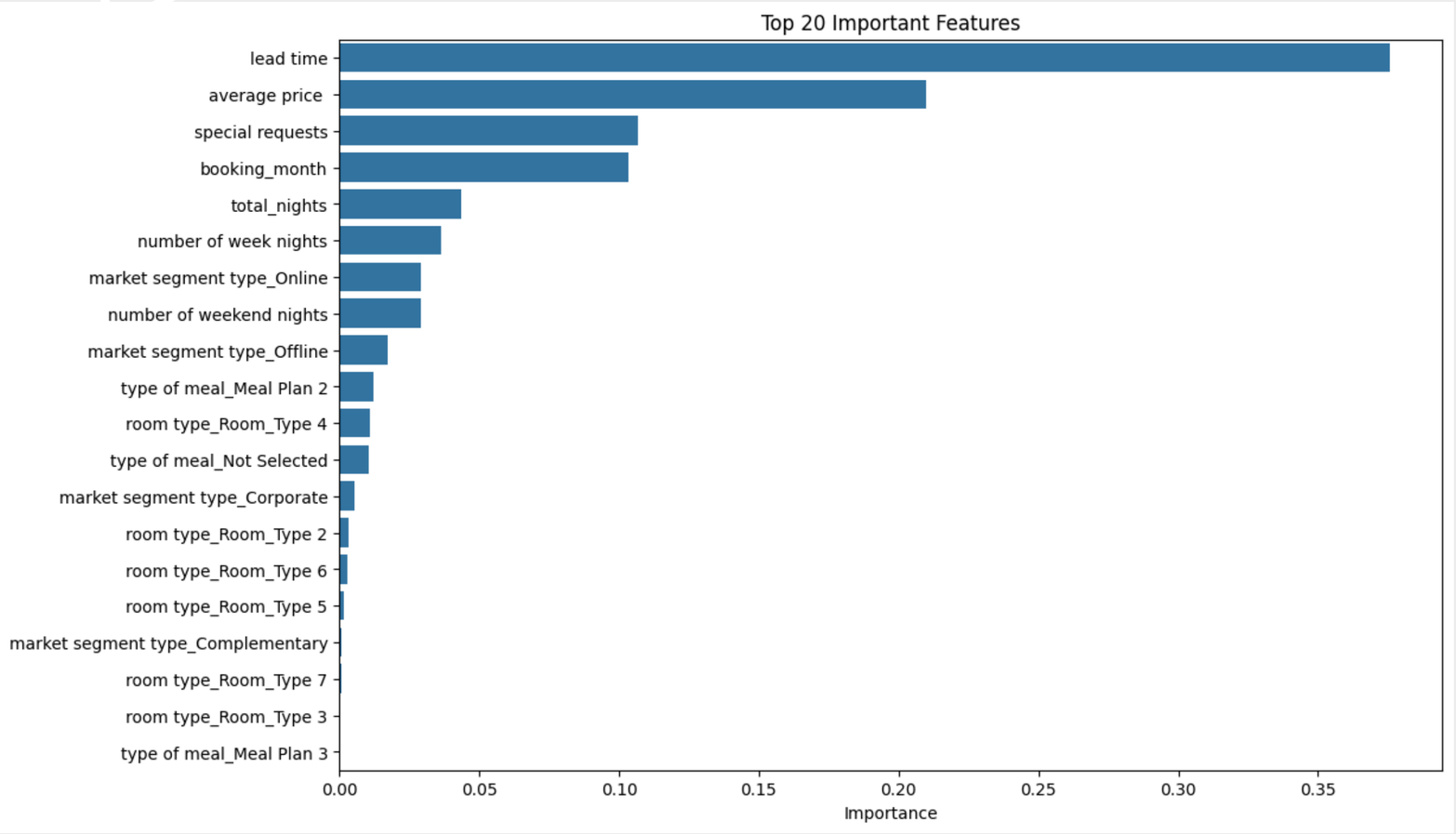
Overall accuracy = 0.89
Interpretation
Random Forest improves accuracy to 89% compared with the logistic baseline
Model recalls 93% of non-canceled bookings and 80% of canceled ones
Balanced precision-recall with weighted F1 of 0.89 shows strong, reliable performance

MODEL COMPARISON (ACCURACY ONLY)

Random Forest delivers the highest accuracy at 0.89
XGBoost follows closely with 0.883
Logistic Regression lags behind at 0.80 and raised a convergence warning; scaling features or increasing max_iter could improve it

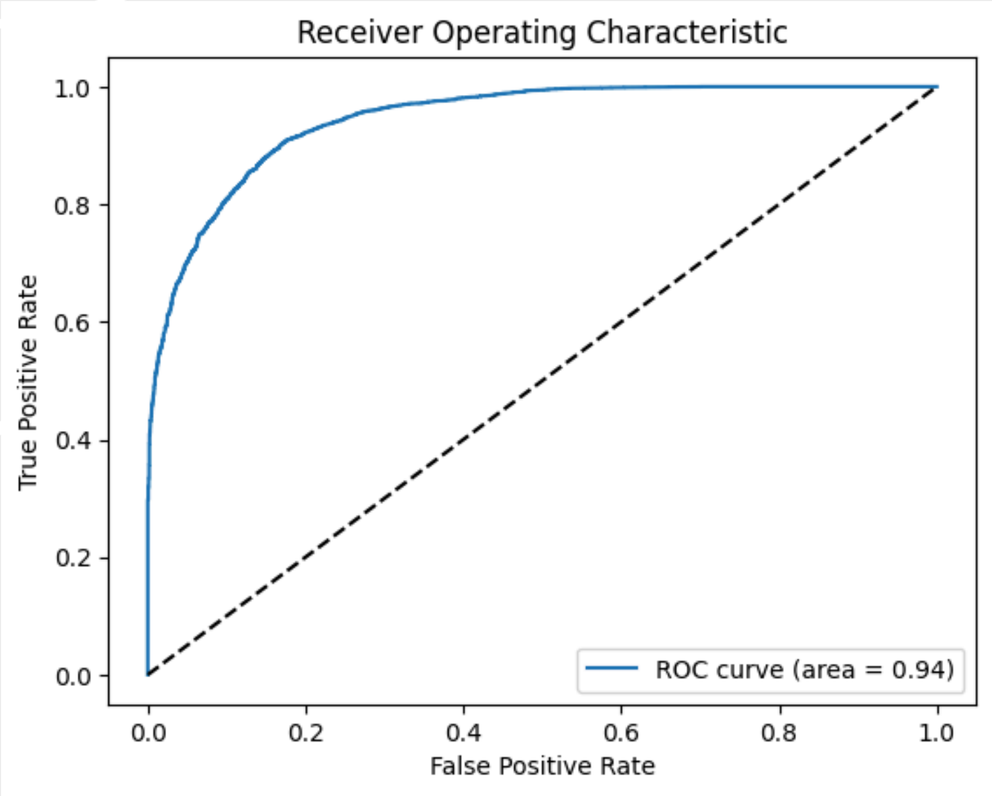


FEATURE IMPORTANCE ANALYSIS



This bar chart shows the most influential features that affected the model's ability to predict booking cancellations. The feature importance was calculated using the trained Random Forest model. Features like lead_time, total_nights, and average_price appeared to have the strongest impact on the model’s decisions. Understanding these important variables helps in improving decision-making and focusing on what truly affects cancellations.

ROC CURVE OF XGBOOST



We evaluated our best model, XGBoost, using the ROC Curve. The curve represents the relationship between the True Positive Rate and the False Positive Rate across different threshold values. The Area Under the Curve (AUC) is 0.94, indicating the model performs very well in distinguishing between canceled and not canceled bookings. AUC values closer to 1 reflect higher classification power. The shape of the curve rises steeply, which means the model detects most cancellations while keeping false alarms low.

THANK YOU

www.reallygreatsite.com