Predictive Analysis On Revenue Per Available Room

The AirBnB Approach

Team 21

Anto Frederic Henry Mohan dass
Gautam Raghu
Rahul Kunku
Sai Mona Duvvapu

Agenda

01 **Introduction**

Marketshare, Value chain

02 **Problem Statement & Data**

Summary Statistics

03 **Preprocessing**

Data Cleaning, Feature Selection, Imputation

04 **Model Training**

Linear/Logistic Regression

05 **Model Validation**

Results, Train/Test, Specificity, Sensitivity

06 **Implications**

Real-world meaning and value

Introduction

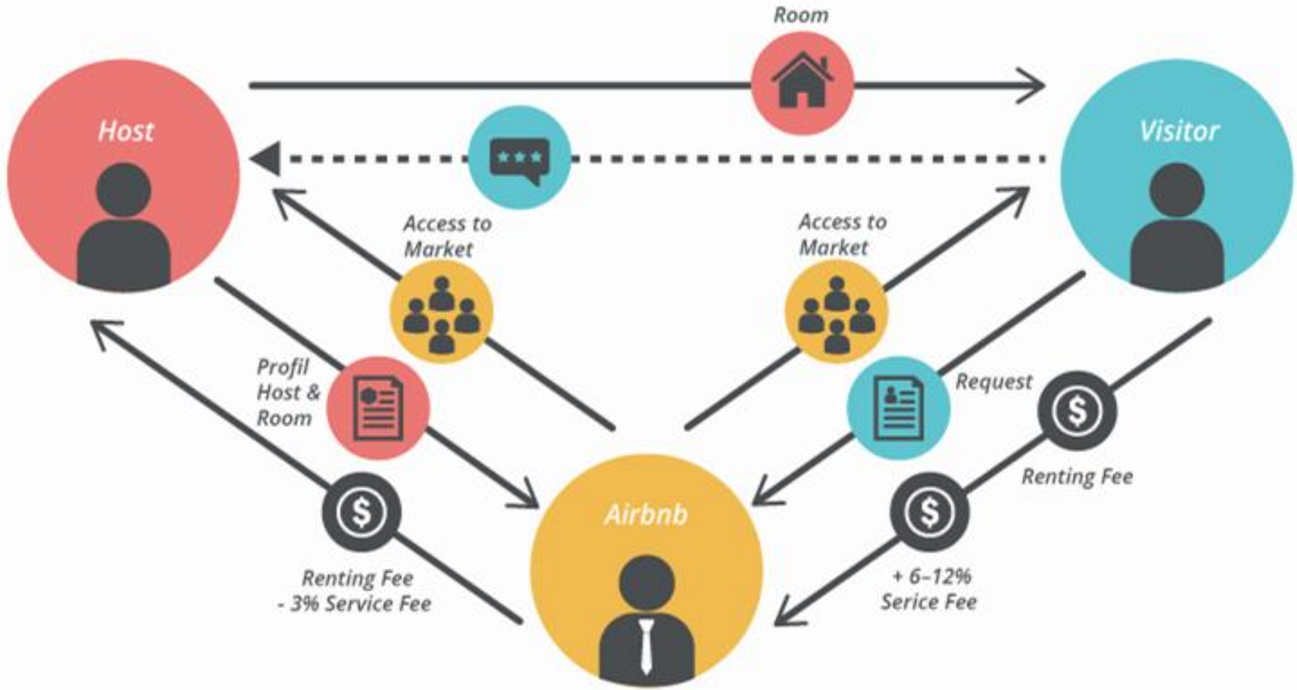
4 Million Hosts

150 Million Users

$3.3 Million Revenue

30% Market share

Value Chain



Platform Business

Problem Statement

Objective

Develop a predictive model using the Airbnb Dallas Dataset

Aim

Accurately predict **RevPAR** to empower hosts in optimizing strategies

Significance

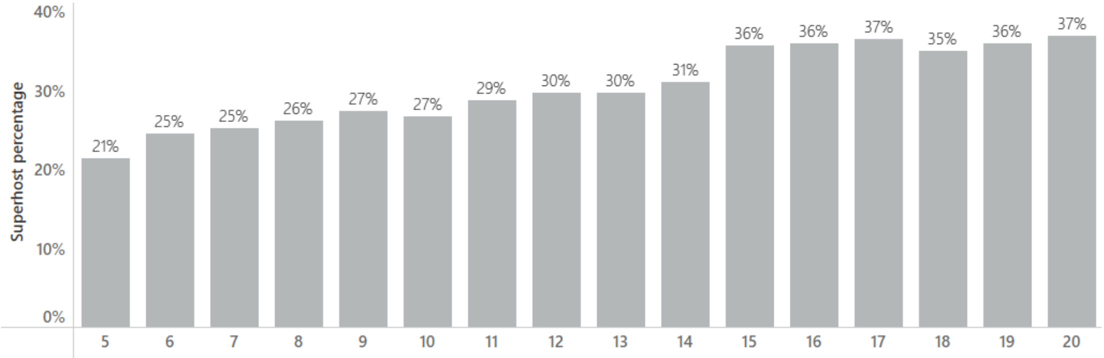
Empower hosts with actionable insights for informed decisions and proactive adjustments.

About the Dataset

Dallas, TX

4,490

9,599

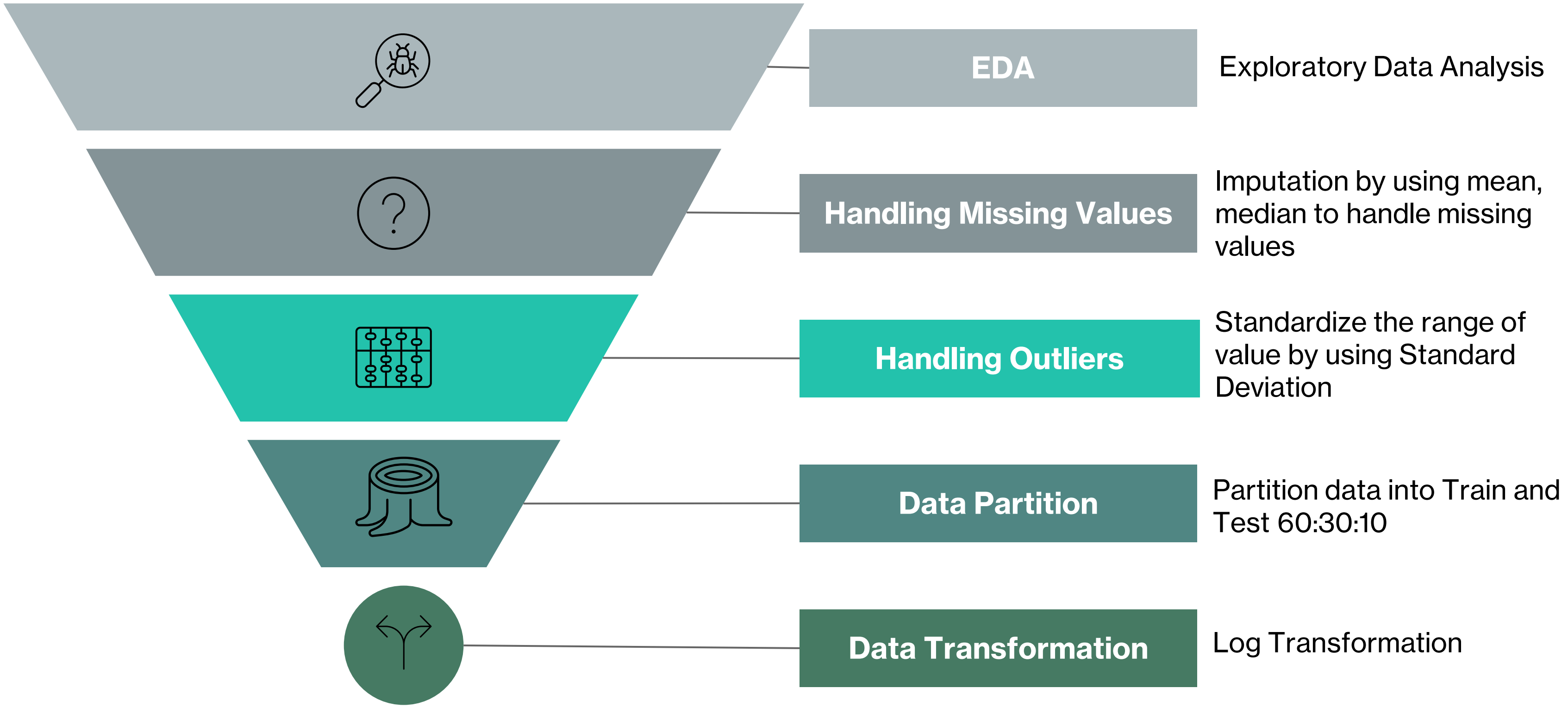


Listing Type	Properties	Avg. Nightly Rate	Avg. Discount	Avg. Occupancy Rate	Avg. rating
Entire home/apt	7,245	186	18.5%	20.6%	4.75
Private room	2,029	72	22.6%	18.7%	4.81
Shared room	302	39	27.5%	15.1%	4.69
Hotel room	23	180	33.2%	17.6%	4.78

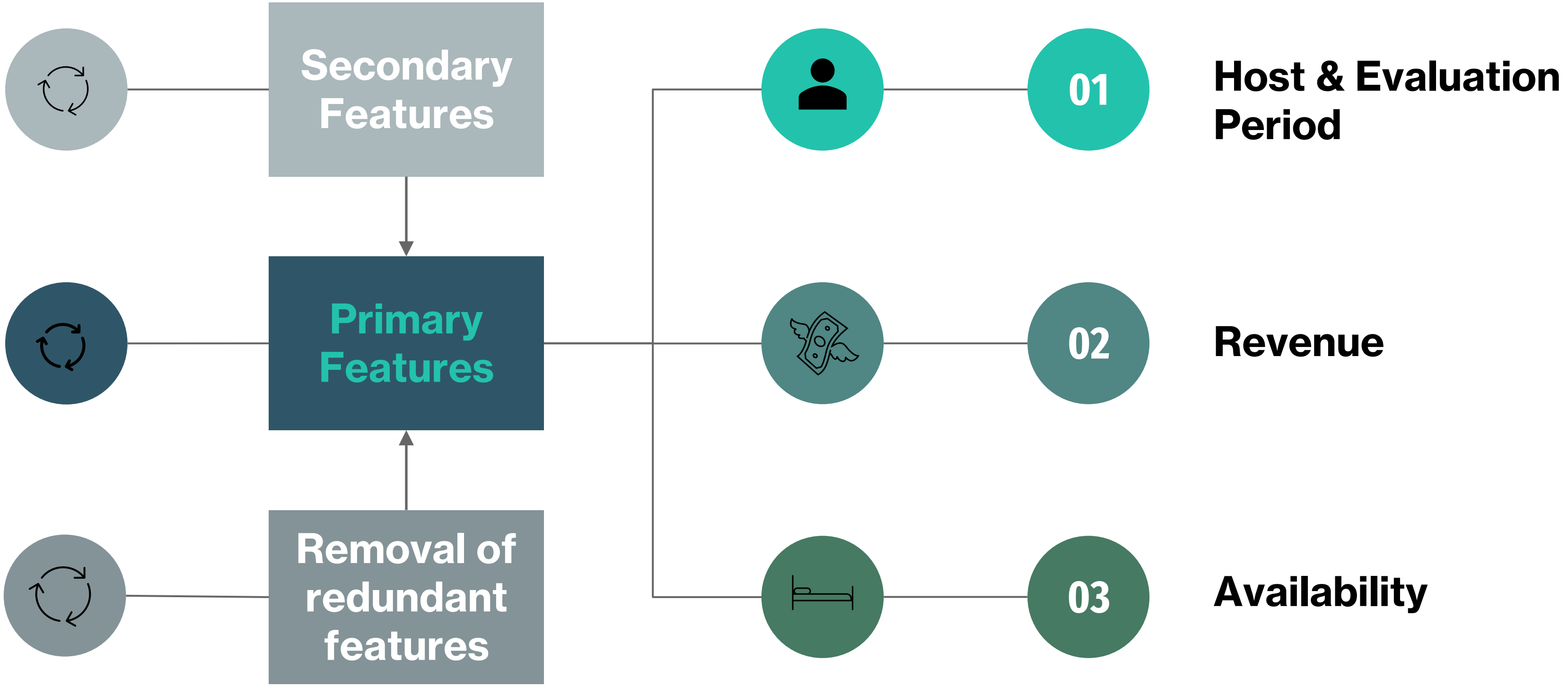
RevPAR?


5
Reasons WHY
01
Performance Benchmarking
02
Pricing Strategy Optimization
03
Occupancy Rate Insights
04
Dynamic Pricing Strategy
05
Predictive Maintenance and Property Upkeep

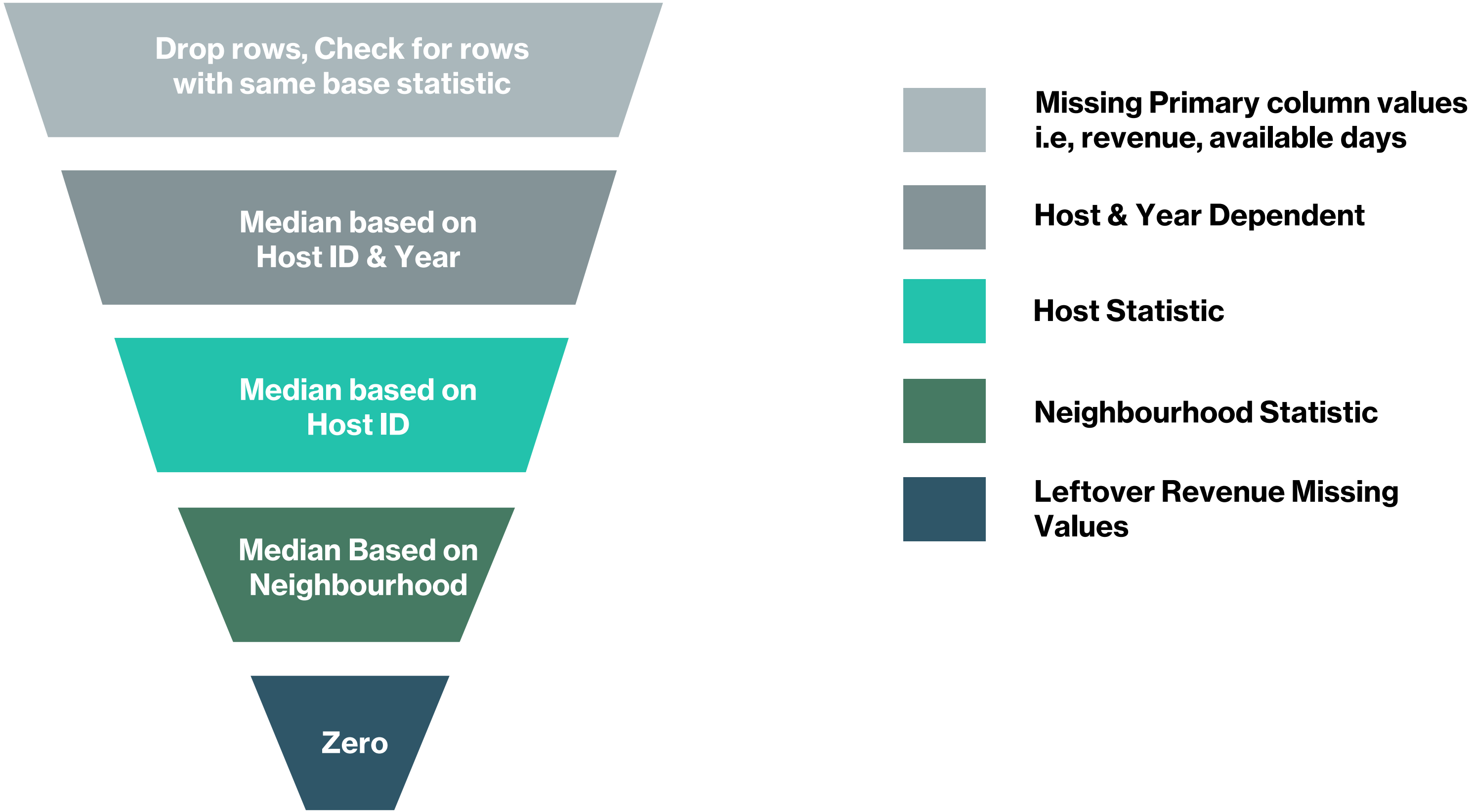
Data Preprocessing



Feature Selection



Missing Values



Outliers

Activity			
01	Data	Data after filling in the missing values	
02	Outliers 1	Filtered values that are above 99 percentile	
03	Outliers 2	Removed values that are beyond the maximum and minimum bound based on boxplot	
04	Outliers 3	Removed values that are beyond +3 and -3 Standard Deviation from the mean	
Processed Data			

Aggregation of Data: Host & Evaluation Period

Data aggregated at host level for each evaluation period before modelling

By Mean

rating_ave_pastYear
numCancel_pastYear
numReviews_pastYear
prop_5_StarReviews_pastYear
available_days_aveListedPrice
booked_days_avePrice
Bedrooms
Bathrooms
Number of Photos
Nightly Rate
Number of Reviews
Rating Overall
occupancy_rate

By Sum

superhost_period_all
numReserv_pastYear
available_days
booked_days
Cleaning Fee (USD)
revenue

Binning

Listing Type
Maximum Guests
Minimum Stays

Modelling

Regression		
01	02	03
Polynomial Regression	**Polynomial Regression**	**Polynomial Regression**
Stepwise without log transformed variables	Without step-wise but on log transformed variables	With Stepwise on log transformed variables

Results

Bathrooms*_1_2_Guests: **0.819**

num_properties_private*num_properties_private: **0.6363**

available_days_aveListedPrice*num_properties_private: **0.0405**

_7__Guests*superhost_percentage: **0.0229**

Number_of_Reviews*Rating_Overall: **0.00154**

Number_of_Reviews*available_days_aveListedPrice: **0.00059**

Implications

01

Operational Efficiency

Identify peak and off-peak periods

02

Resource Allocation

Justify investments in the resources

03

Competitive Analysis

To identify market trends

04

Loyalty Programs

To repeat customers

05

Investor and Stakeholder Relations

Performance Indicator

Hosts

THANK YOU

What is RevPAR?

Mathematically calculated as:

$$\frac{\text{Total Revenue}}{\text{Available Rooms}}$$

Selected Features

- 'rating_ave_pastYear’
- 'numCancel_pastYear’
- 'numReviews_pastYear'
- 'prop_5_StarReviews_pastYear’
- 'available_days_aveListedPrice'
- 'booked_days_avePrice’
- 'Bedrooms’
- 'Bathrooms’
- 'Number of Photos'
- 'Nightly Rate’
- 'Number of Reviews’
- 'Rating Overall’
- 'occupancy_rate'

- numReserv_pastYear'
  'available_days'
- 'booked_days’
- 'Cleaning Fee (USD)’
- 'superhost_percentage’
- 'num_properties_home’
- 'num_properties_hotel'
- 'num_properties_private’
- 'num_properties_shared’
- 'num_properties_stay_1-2_days’
- 'num_properties_max_3-10_days’
- 'num_properties_max_10+_days'

Code Snippets – Missing Variables

```python
# Assuming your dataset is named 'airbnb_data'
# Fill NaN values in 'Neighborhood' based on associated zip codes
df['Neighborhood'] = df.groupby('Zipcode')['Neighborhood'].transform(lambda x: x.fillna(x.mode().iloc[0]))

# Verify if NaN values in 'Neighborhood' have been replaced
missing_neighborhoods = df[df['Neighborhood'].isnull()]

# If there are still missing values, check the unique Zipcodes with NaN Neighborhoods
missing_zipcodes = missing_neighborhoods['Zipcode'].unique()

# Fill NaN values in 'Neighborhood' based on common Zipcodes
for zipcode in missing_zipcodes:
    common_neighborhood = df.loc[df['Zipcode'] == zipcode, 'Neighborhood'].dropna().unique()
    df.loc[(df['Zipcode'] == zipcode) & (df['Neighborhood'].isnull()), 'Neighborhood'] = common_neighborhood[0]

# Verify if all NaN values in 'Neighborhood' have been replaced
final_missing_neighborhoods = df[df['Neighborhood'].isnull()]
```

```python
# Replace missing values within each 'Airbnb Host ID' and "Year'"
df['Rating Overall'] = df.groupby(['Airbnb Host ID','Year'])['Rating Overall'].transform(replace_missing_with_median)

# Replace missing values within each 'Airbnb Host ID'
df['Rating Overall'] = df.groupby(['Airbnb Host ID'])['Rating Overall'].transform(replace_missing_with_median)

# Replace missing values within each 'Neighbourhood'
df['Rating Overall'] = df.groupby(['Neighborhood'])['Rating Overall'].transform(replace_missing_with_median)
```

```python
# Replace na values with the mean of the non-na values of the particular host ID and year
def replace_missing_with_median(group):
    non_null_values = group.dropna()  # Filter non-null values
    if non_null_values.empty:
        return group  # Return as is if no non-null values present
    else:
        median_val = non_null_values.median()  # Calculate median of non-null values
        return group.fillna(median_val)  # Fill missing values with median
```

Code Snippets – SAS EM

Property	Value
General	
Node ID	Part
Imported Data	...
Exported Data	...
Notes	...
Train	
Variables	...
Output Type	Data
Partitioning Method	Default
Random Seed	59911
Data Set Allocations	
Training	60.0
Validation	30.0
Test	10.0
Report	
Interval Targets	Yes
Class Targets	Yes
Status	
Create Time	12/8/23 3:40 PM
Run ID	[illegible]

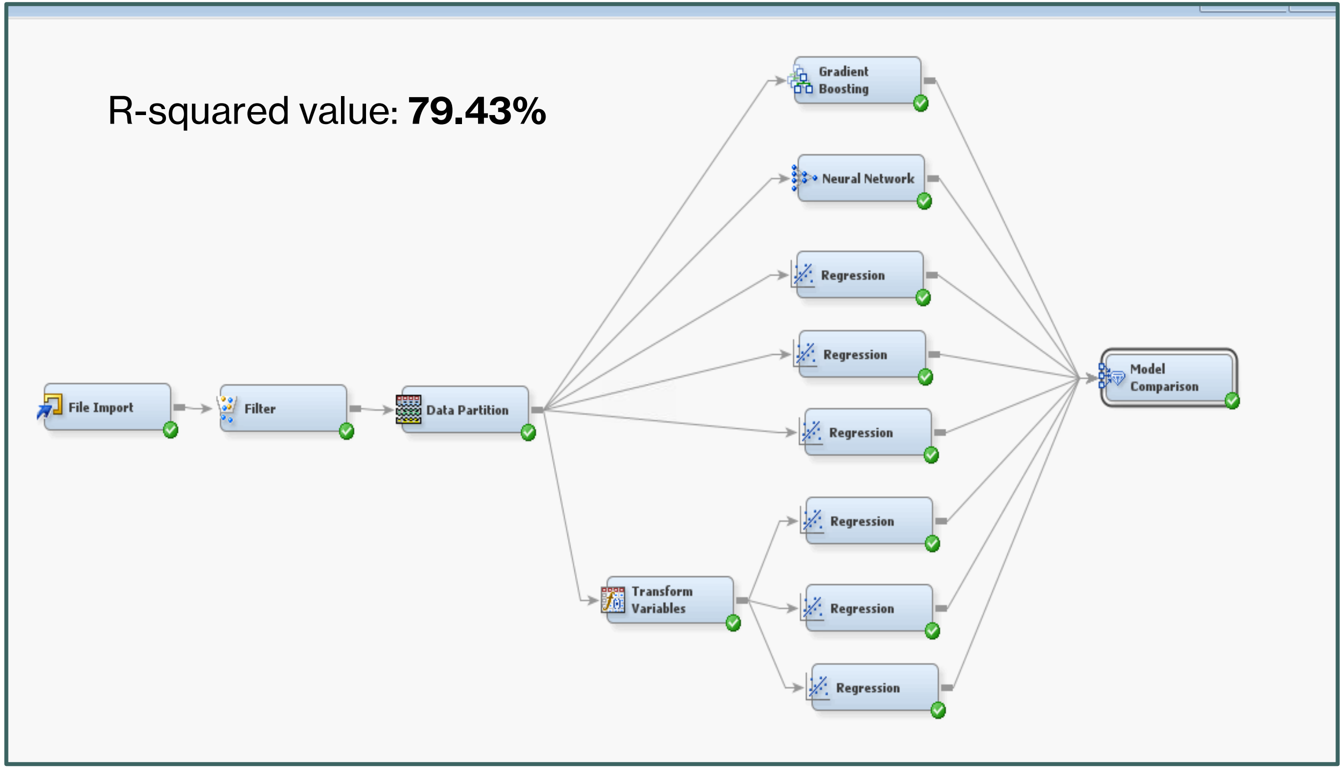
General

Property	Value
Variables	...
Formulas	...
Interactions	...
SAS Code	...
Default Methods	
Interval Inputs	Log
Interval Targets	None
Class Inputs	None
Class Targets	None
Treat Missing as Level	No
Sample Properties	
Method	First N
Size	Default
Random Seed	12345
Optimal Binning	
Number of Bins	4
Missing Values	Use in Search
Grouping Method	
Cutoff Value	0.1
Group Missing	No

Property	Value
Export Table	Filtered
Tables to Filter	Training Data
Distribution Data Sets	Yes
Class Variables	
Class Variables	...
Default Filtering Method	Rare Values (Percentage)
Keep Missing Values	Yes
Normalized Values	Yes
Minimum Frequency Cutoff	1
Minimum Cutoff for Percentage	0.01
Maximum Number of Levels Cutoff	25
Interval Variables	
Interval Variables	...
Default Filtering Method	Standard Deviations from the Mean
Keep Missing Values	Yes
Tuning Parameters	...
Score	
Create Score Code	Yes
Update Measurement Level	No
Status	

Results

R-squared value: **79.43%**



References

- https://6sense.com/tech/reservation-and-online-booking/airbnb-market-share
- https://bmtoolbox.net/stories/airbnb/
- https://www.investopedia.com/
- https://chat.openai.com/