Project Name: Airbnb Success Predictor

Team Members: Apoorva Shastry, Ananya Praveen Shetty

Date: September 7, 2025

1.Business Purpose & Use Case

Business Purpose: To create a machine learning model that gives data-driven advice for real estate investors interested in the short-term rental market. The model will help investors find properties with the best chances for profitability and occupancy, reducing the risk of a bad investment.

Business Use Case: An investor is thinking about buying a 2-bedroom condo in Austin, TX,SF etc to list on Airbnb. Before making a significant investment, they need to address key questions:

- What is the realistic nightly price I can charge for this property?
- How often can I expect the property to be booked (i.e., what is the occupancy rate)?
- What property features (amenities, location, etc.) are most important for success in this market?
- How does this potential property compare to other high-performing listings in the area?
- This project aims to develop a tool that answers these questions, turning a high-stakes guess into a well-informed business decision.

2. Goals

1. **Predict Nightly Price**: Build a regression model to predict the best nightly price for a potential Airbnb listing.
2. **Forecast Occupancy**: Create a classification model to check if a property is likely to be "high-demand," meaning it will have a high occupancy rate.
3. **Identify Key Drivers**: Analyze the model results to find the key features, such as specific amenities and location, that contribute to a successful rental.
4. **Develop a "Success Score"**: Create a score that ranks potential investment properties based on their predicted price and demand. This score will give investors a clear metric to work with.

3. Data Narrative & Acquisition

- **Data Narrative:** The basis of this project is real-world listing data from Airbnb. This data provides a snapshot of the short-term rental market in a major city. It includes various features for each property, such as its location, price, characteristics, and details provided by the host. We will enhance this main dataset with external sources to add important context about neighborhood appeal and local economics.

- **Datasets:**
    1. **Initial Dataset: Inside Airbnb**. We will use the listings.csv file for a major metropolitan area like Austin, TX, or Los Angeles, CA. This dataset contains thousands of listings with dozens of features like price, latitude, longitude, room_type, accommodates, amenities, and review_scores_rating.
        - *Source:* https://insideairbnb.com/get-the-data/
    2. **Second Dataset: U.S. Census Bureau Data.** To understand the socioeconomic context of each listing's neighborhood, we will use American Community Survey (ACS) data. We will join this data by census tract or ZIP code to add features like median_income, population_density, and unemployment_rate.
        - *Source:* https://data.census.gov/
    3. **Third Dataset: Walk Score API or Dataset.** Proximity to amenities is a key driver of rental success. We will use a Walk Score dataset or its API to generate a "walkability" score for each property, quantifying its closeness to restaurants, parks, and public transit.
        - *Source:* https://www.walkscore.com/professional/walk-score-data.php

---

4. Research Component

- **Topic 1: Dynamic Pricing Strategies:** How do hosts adjust pricing based on seasonality, local events, and demand?

    https://www.researchgate.net/publication/390827428_Dynamic_Pricing_and_Seasonality_Insights_From_Short-Term_Rental_Market

    https://www.revoptimum.com/blog/the-impact-of-local-events-on-hotel-revenue-how-to-capitalize-on-seasonal-demand

    Airbnb pricing strategy:
    https://www.sciencedirect.com/science/article/abs/pii/S0278431922002584?via%3Dihub

- **Topic 2: The "Guest Experience":** Which amenities and host behaviors correlate most strongly with positive reviews?

    determinants of airbnb guest satisfaction:https://www.nature.com/articles/s41598-024-75701-w impact of amenities on hotel reviews:https://www.tandfonline.com/doi/full/10.1080/1528008X.2020.1814935

online reviews trust short term rentals: https://www.sciencedirect.com/science/article/pii/S2444883423000232

- **Topic 3: Impact of Local Regulations:** How do city ordinances (e.g., taxes, licensing, zoning) affect the profitability of short-term rentals?

  impact of short-term rental regulations los angeles:https://cepr.org/voxeu/columns/short-term-rentals-and-housing-market-quasi-experimental-evidence-airbnb-los-angeles

  airbnb housing market zoning:https://www.sciencedirect.com/science/article/abs/pii/S0166046221000272

  economic impact of rental restrictions: https://www.brookings.edu/articles/what-does-economic-evidence-tell-us-about-the-effects-of-rent-control/

---

5. Machine Learning Experiments

Our experiments are designed to answer the core business questions of our investor use case.

Experiment 1: Regression (Price Prediction)

- **Business Question:** *What is the optimal nightly price I can charge for a new property?*
- **ML Approach:** We will train a regression model (e.g., XGBoost, Random Forest) to predict a listing's price.
- **Target Variable:** price
- **Features:** accommodates, bedrooms, bathrooms, latitude, longitude, room_type, one-hot encoded amenities.
- **Success Metric:** Mean Absolute Error (MAE) to give us an easily interpretable "dollar amount" error.

Experiment 2: Classification (Demand Prediction)

- **Business Question:** *Is this property likely to be a "high-demand" listing with high occupancy?*
- **ML Approach:** We will engineer a target variable is_high_demand and train a classification model (e.g., Logistic Regression, SVM). A property is "high-demand" if it is booked for more than 75% of the year.
- **Target Variable:** is_high_demand (Binary: 1 if availability_365 < 90, 0 otherwise).

- **Features:** review_scores_rating, number_of_reviews, predicted_price (from our regression model), walk_score.
- **Success Metric:** F1-Score to balance precision and recall.

Experiment 3: Clustering (Market Segmentation) (Assignment done with K-means clustering)

- **Business Question:** *Which properties are in the "golden cluster" of being highly-rated and profitable but are not in the most obvious tourist-heavy locations?*
- **ML Approach:** We will use K-Means and DBSCAN to segment the listings into distinct market groups.
- **Features for Clustering:** latitude, longitude, price, review_scores_rating.
- **Analysis:** We will analyze the resulting clusters to identify "hidden gems"—clusters with high average prices and ratings located outside the primary downtown or tourist cores. This provides strategic insights into emerging, high-potential neighborhoods.

---

6. Key Performance Indicators (KPIs)

To measure the success of a potential investment, we will calculate two primary KPIs:

1. **Projected Annual Revenue:** This is the ultimate financial metric for an investor.
   - **Formula:** Projected Annual Revenue = (Predicted Price) * (365 * Predicted Occupancy Rate)
   - *Note: Predicted Occupancy Rate will be derived from our classification model's output.*
2. **Property Success Score:** A single, normalized score from 0-100 for easy comparison.
   - Formula: A weighted average of our model outputs:
     Score = 0.5 * (Normalized Price) + 0.3 * (High-Demand Probability) + 0.2 * (Normalized Review Score)

Assignment 1: Project Plan & Research

1. Business Purpose & Use Case

The primary business objective of this project is to create a machine learning model that provides data-driven guidance for real estate investors in the short-term rental market. The model aims to identify properties with the highest potential for profitability and occupancy, thereby minimizing investment risk.

The core use case involves an investor considering a property purchase in Los Angeles. Before investing, they need answers to critical questions: What is the optimal nightly price? What is the likely occupancy rate? Which features are most critical for success? Our project's mission is to answer these questions, turning a high-stakes guess into a calculated business decision.

2. Goals

1. **Predict Nightly Price:** Build a regression model to predict the optimal price.
2. **Forecast Occupancy:** Create a classification model to predict "high-demand" properties.
3. **Identify Key Drivers:** Use model results to identify the most influential features.
4. **Develop a "Success Score":** Create a composite score to rank investment opportunities.

3. Data Strategy & Research

Our project is built on three datasets: **Airbnb Listings**, **U.S. Census Income Data**, and a **Walkability Index**. To ground our work, we also researched academic papers on Dynamic Pricing, the Guest Experience, and the Impact of Local Regulations on the short-term rental market.

Assignment 2: Unsupervised Learning - Fractal Clustering

The first analytical phase of our project was to use unsupervised learning to segment the LA market and identify a "golden cluster" of high-potential properties.

1. Fractal Clustering Methodology

We implemented an iterative "Fractal Clustering" approach to refine our search for the best properties.

- **Objective Functions:** To quantitatively define our target, we wrote two objective functions:
    1. **Profitability Score:** `(Average Price) * (Average Rating)^2` — This rewards clusters that are both lucrative and high-quality.
    2. **Quality Score:** `Average Rating` — This focuses purely on guest satisfaction.
- **Performance Metrics:** To measure the quality of our clusters, we computed the **Sum of Squared Errors (SSE)** for compactness and the **Silhouette Score** for cluster separation.

2. Analysis and The Golden Cluster

- **Iteration 1:** We ran K-Means on the entire dataset of ~27,000 listings. The model produced five distinct market segments, with a Silhouette Score of **0.2994**.

Iteration 1 Results

Cluster	Avg Price	Avg Rating	Listing Count	Profitability Score
2	$464.86	4.87	3,321	11020.3
3	**$167.38**	**4.83**	**3,768**	**3898.0**
0	$148.76	4.82	13,660	3459.1
4	$127.84	4.81	5,723	2962.3
1	$144.78	2.57	539	958.5

While the luxury "Coastal Elite" cluster (Cluster 2) had the highest raw profitability, we made a strategic business decision to select the next-best group as our initial **Golden Cluster (Cluster 3)**. This cluster represented the "sweet spot" of the market: a segment of **3,768 listings** with a high average price (**$167**) and an excellent rating (**4.83**), making it the most strategic and accessible target for an investor.

- **Iteration 2:** We then re-clustered this "golden" subset. This fractal step successfully isolated a more elite sub-cluster of **189 listings** with a much higher average price of **$360**. This became our final, most refined **Golden Cluster**.

Iteration 2 Results (on Cluster 3 subset)

Sub-Cluster	Avg Price	Avg Rating	Listing Count	Profitability Score
1	**$360.56**	**4.70**	**189**	**7964.3**
2	$165.24	4.83	1,767	3849.5
0	$149.31	4.84	1,812	3495.1

- **Outcome:** The primary outcome of this assignment was the creation of a new binary column in our dataset: **`is_golden_cluster`**. The 189 listings in our final golden cluster (Sub-Cluster 1 from Iteration 2) were labeled '1', and all others were labeled '0'. This created the perfect target variable for our next assignment on classification.

Assignment 3: Supervised Learning - Amalgamation & Classification

The final phase of the project focused on enriching our data through amalgamation and training supervised learning models to predict the "golden cluster" properties.

1. Amalgamation Methodology

We performed a sequential, two-phase amalgamation:

1. **Phase 1 (Creating `Dataset 1+2`):** We performed **Reverse Geocoding** on the Airbnb listings to generate a `zip_code` for each. We then used an **Attribute Left Join** to merge the U.S. Census `median_income` data.
2. **Phase 2 (Creating `Dataset 1+2+3`):** We then performed a **Spatial Left Join** using `GeoPandas`. This process merged the `Walkability` score by identifying which Census Tract polygon each Airbnb's coordinate point was located `within`.

In all steps, **left joins** were used to ensure no original Airbnb listings were lost.

2. The "Muller Loop": Classification Experiments

To test the impact of our amalgamation, we conducted a "Muller Loop," running multiple classification algorithms on each of the three incrementally amalgamated datasets. Each team member tested a different set of algorithms. The goal was to predict the `is_golden_cluster` label we created in the previous assignment. We focused on the **F1 Score** and **AUC** as our primary metrics due to the highly imbalanced nature of the data.

3. Results and Performance Analysis

The combined results from both team members' experiments are summarized below.

Ananya's Analysis (Ensemble Models)

Experiment	F1 Score	AUC
Dataset 1+2+3 - Random Forest	**0.9798**	0.9952
Dataset 1+2+3 - XGBoost	**0.9738**	0.9992

Apoorva's Analysis (Instance-Based and Boundary Models)

Experiment	F1 Score	AUC
Dataset 1+2+3 - KNN	0.8855	0.9741
Dataset 1+2+3 - SVM	0.7585	0.9645

The results show a clear trend: the effectiveness of amalgamation is highly dependent on the model's complexity.

- The advanced ensemble models (**Random Forest and XGBoost**) benefited from the enriched data, achieving their **highest F1 Scores on the fully amalgamated** `Dataset 1+2+3`.
- Conversely, the performance of the simpler, distance-based models (**KNN and SVM**) either stayed flat or slightly **decreased**, suggesting the added dimensions introduced complexity that these algorithms could not handle as effectively.

4. Overall Project Conclusion

This project successfully demonstrates the power of a multi-stage machine learning workflow. Through **unsupervised clustering**, we were able to define a meaningful, data-driven target for what constitutes a "golden" investment property.

Subsequently, through **data amalgamation and supervised classification**, we proved that enriching a dataset with relevant socioeconomic and locational context **measurably improves the performance of advanced models**. The key takeaway is twofold: data amalgamation is a critical technique for building high-performance models, but its value is fully realized only when paired with an algorithm sophisticated enough to uncover the complex, non-linear patterns within the newly added features.

Assignment 4: Interactive Dashboard - Data Distribution Impact on Model Performance

Objective

The fourth phase of our project investigated how data distribution manipulation affects machine learning model performance in imbalanced classification scenarios. We built an interactive dashboard to dynamically adjust class balance through resampling techniques and observe real-time impacts on model metrics.

Methodology

1. Resampling Strategies Implementation

We implemented three distinct sampling approaches to address the severe class imbalance (15% golden cluster vs 85% non-golden):

- **Undersampling (0-49% slider range)**: Used RandomUnderSampler to reduce majority class samples while preserving all minority class instances

- **Original Distribution (50% slider position)**: Maintained natural class balance as baseline comparison
- **SMOTE Oversampling (51-100% slider range)**: Applied Synthetic Minority Over-sampling Technique to generate synthetic minority class samples using k-nearest neighbors interpolation

2. Multi-Algorithm Comparison

We expanded beyond our previous ensemble focus to test four distinct algorithm families:

Algorithm	Baseline F1 Score	Architecture Type
XGBoost	0.9842	Gradient Boosting Ensemble
Random Forest	0.9819	Bootstrap Aggregating Ensemble
MLP (Neural Network)	0.9692	Deep Learning
SVM	0.7428	Support Vector Machine

3. Feature Importance Re-analysis

Using Random Forest feature importance on the amalgamated dataset, we identified the key predictors:

1. **price** (54.91%) - Dominant predictor due to target definition
2. **review_scores_rating** (37.88%) - Quality indicator independent of price
3. **minimum_nights** (3.35%) - Booking policy proxy
4. **room_type** (2.46%) - Property type classification
5. **Walkability** (1.40%) - Location accessibility (surprisingly low impact)

Interactive Dashboard Components

The dashboard featured three control mechanisms:

1. **Feature Selection Dropdown**: Choose which numerical feature's distribution to modify
2. **Sampling Intensity Slider**: Adjust resampling ratio from 0% (aggressive undersampling) to 100% (maximum SMOTE oversampling)
3. **Algorithm Selection**: Switch between XGBoost, Random Forest, MLP, and SVM

For each configuration, the system displayed five integrated visualizations:

- Confusion matrix heatmap
- ROC curve with AUC metric
- Specificity vs Sensitivity comparison
- F1 score baseline comparison
- Feature distribution histogram

Key Findings

1. Algorithm-Specific Resampling Sensitivity

The impact of resampling varied dramatically by algorithm complexity:

- **Tree-based methods (XGBoost, RF)**: Minimal performance change across all sampling ratios (-0.5% to +1% F1 variance). These algorithms demonstrated inherent robustness to class imbalance through their splitting criteria and ensemble nature.

- **Neural Network (MLP)**: Moderate improvement with balanced data (+3-5% F1 increase at 60-70% sampling). The gradient descent optimization benefited from more balanced gradient updates.

- **Support Vector Machine (SVM)**: Most dramatic improvement (+10-15% F1 increase). The maximum margin optimization, which typically favors majority class in imbalanced scenarios, achieved substantially better minority class recall with balanced data.

2. Optimal Resampling Configuration

Through systematic experimentation, we identified the optimal strategy:

- **Technique**: SMOTE oversampling
- **Intensity**: 60-70% slider position
- **Resulting balance**: Approximately 2.5:1 to 3:1 majority-to-minority ratio
- **Rationale**: Preserves all original data while providing sufficient synthetic examples to improve minority class representation without overfitting risk

3. Specificity-Sensitivity Tradeoffs

Analysis of the specificity vs sensitivity plots revealed:

- **Baseline (original distribution)**: High specificity (>0.99) but lower sensitivity (0.67-0.97 depending on algorithm)
- **With SMOTE at 70%**: Maintained specificity (>0.97) while improving sensitivity (+5-15% for weaker models)
- **Business implication**: Balanced approach successfully identifies more golden cluster properties without significantly increasing false positives

Performance Comparison Across Sampling Strategies

Configuration	XGBoost F1	Random Forest F1	MLP F1	SVM F1
Undersample (30%)	0.9820	0.9800	0.9750	0.7980
Original (50%)	0.9842	0.9819	0.9692	0.7428
SMOTE (70%)	0.9838	0.9825	0.9810	0.8650

Technical Insights

Challenge Encountered: The median_income feature from Assignment 3's amalgamation showed 100% missing values after ZIP code merge, requiring exclusion from the final model. This highlighted the importance of data quality validation in multi-source integration.

Visualization Approach: After encountering rendering issues with Plotly in the ipywidgets context, we pivoted to matplotlib/seaborn for reliable real-time visualization updates within the interactive dashboard.

Conclusions and Recommendations

For Production Deployment:

1. **Recommended Algorithm**: XGBoost with SMOTE at 65% sampling
    - Achieves highest baseline performance

    - Minimal computational overhead
    - Robust across distribution variations
2. **Alternative for Interpretability**: Random Forest with original distribution

    - Minimal performance gap from XGBoost
    - Provides clear feature importance rankings
    - No resampling complexity in production pipeline
3. **Not Recommended**: SVM despite improvement with resampling

    - Baseline performance significantly below ensemble methods
    - Computational expense for marginal final performance
    - Requires careful hyperparameter tuning

Strategic Insight: This assignment demonstrated that data distribution manipulation is most valuable when paired with algorithms that struggle with imbalance (SVM, MLP), while advanced ensemble methods (XGBoost, Random Forest) achieve excellent performance even on imbalanced data. For our golden cluster prediction use case, the inherent robustness of gradient boosting algorithms makes complex resampling strategies optional rather than essential.

Integration with Overall Project

This assignment completed our supervised learning pipeline by:

1. Validating the robustness of our Assignment 3 classification results across multiple algorithms
2. Demonstrating that our golden cluster definition (from Assignment 2) is consistently learnable across different model architectures
3. Establishing production-ready configurations for real-world deployment in the investor decision support tool

The interactive dashboard serves as both an analytical tool and a demonstration platform, allowing stakeholders to understand model behavior under different data conditions and build confidence in the system's recommendations.

Of course. I have analyzed the files you provided for Week 6 and Week 7.

It appears that the content from your "Week 6" file (Interactive Dashboard for Data Distribution Analysis) is the same as the "Assignment 4" section you already included in your original project description.

Here is the new content from your "Week 7" files, formatted to be appended to your project as **Assignment 5** and **Assignment 6**.

Assignment 5: Optimizing 'Golden Cluster' Prediction through Feature Importance

1. Introduction & Objective

The primary objective of this assignment was to demonstrate a measurable improvement in our 'Golden Cluster' classification models by strategically removing noisy and irrelevant features. Building on the amalgamated dataset, this analysis uses data-driven feature selection to create a more streamlined and powerful predictive model.

Our hypothesis was that the 'Walkability' feature, despite being intuitively relevant, introduced more noise than signal for this specific classification task.

2. Methodology

The analysis followed a systematic four-step process:

1. **Baseline Performance:** First, we established a baseline by training our suite of classifiers (XGBoost, Random Forest, MLP, SVM) on the complete, five-feature amalgamated dataset (price, review_scores_rating, minimum_nights, room_type, Walkability).
2. **Feature Importance Calculation:** We used a Random Forest Classifier to quantify the predictive contribution of each feature. The results confirmed our hypothesis:
    - price: 54.91%
    - review_scores_rating: 37.88%
    - minimum_nights: 3.35%
    - room_type: 2.46%
    - **Walkability: 1.40%**
3. **Strategic Feature Selection:** Based on its negligible importance score of 1.4%, the Walkability feature was removed.
4. **Comparative Model Retraining:** We retrained all four models on the new, optimized four-feature dataset to compare performance.

3. Results & Business Impact

The removal of the Walkability feature resulted in a universal and measurable improvement in model performance, particularly in the F1 Score, which is critical for our imbalanced dataset9.

Performance Comparison (F1 Score)

Model	Baseline F1 (5 Features)	Optimized F1 (4 Features)	Improvement

XGBoost	0.9842	**0.9855**	+0.13%
Random Forest	0.9819	**0.9831**	+0.12%
MLP (Neural Net)	0.9692	**0.9708**	+0.16%
SVM	0.7428	**0.7585**	+2.11%

Key Business Impacts:

- **Improved Predictive Accuracy:** By removing distracting noise, we created more reliable models for identifying 'golden cluster' properties.
- **Increased Model Efficiency:** The optimized models use 20% fewer features, translating directly to faster training times and lower computational costs for deployment.
- **Streamlined Data Strategy:** This finding simplifies future data collection, as we now know Walkability is not a key predictor for this specific task.

Assignment 6: Latent Features & Regression Analysis for Price Prediction

1. Objective

This assignment shifted focus from classification back to one of our original project goals: Predict Nightly Price. The objective was to investigate whether engineered "latent features" (features derived from combinations of existing data) could improve the performance of our regression models beyond what was possible with raw or simple external data alone.

2. Methodology

We created three progressively enriched feature sets and tested them across a suite of six regression algorithms (including Linear Regression, Random Forest, XGBoost, and a Keras MLP)15.

- **Feature Set 1: Base Features (6 features)**
    1. Raw attributes: accommodates, bathrooms, bedrooms, beds, minimum_nights, room_type.

- **Feature Set 2: Enriched Features (8 features)**
    1. Base Features + External Data: median_income and Walkability.
- **Feature Set 3: Latent Features (11 features)**
    1. Enriched Features + three new engineered features:
    2. **Host Experience Score:** log(host_days_active * host_activity_level)
        - *Rationale: Experienced, active hosts may price more strategically*
    3. **Popularity Score:** log(number_of_reviews * normalized_rating)
        - *Rationale: Popular, highly-rated listings can command premium prices*.
    4. **Space Efficiency Score:** accommodates / (bedrooms + 1)
        - *Rationale: Efficient use of space (e.g., a 1-bed that sleeps 4) may indicate higher value*.

3. Results and Key Findings

The experiment confirmed that feature engineering is critical for improving model performance. The Random Forest model trained on the final Latent Features set was the clear winner.

Overall Performance Comparison (Best Model: Random Forest)

Feature Set	**Best Model**	**RMSE ($)**	**R² Score**	**Improvement (vs. Base)**
Base	Random Forest	$113.73	0.533	(baseline)
Enriched	Random Forest	$111.59	0.550	+3.2%
Latent	Random Forest	**$110.62**	**0.558**	**+4.7%**

Ananya and Apoorva contribution :

Assignment / Week	**Core Task(s)**	**Ananya Praveen Shetty's Contributions**	**Apoorva Shastry's Contributions**
Assignment 1 (Week 1-2)	**Project Plan & Research**	Co-authored the project plan, focusing on defining the "Goals" and "KPI" sections. Researched academic papers on "The Guest Experience" and "Dynamic Pricing" to inform the project's research component.	Co-authored the project plan, defining the primary "Business Purpose & Use Case." Led data acquisition strategy and research, identifying the Census, Inside Airbnb, and Walk Score datasets.
Assignment 2 (Week 3)	**Unsupervised Learning**	Developed the "Profitability Score" objective function. Ran the initial K-Means (Iteration 1) on the full dataset. Analyzed the results to strategically select the initial "Golden Cluster" for the fractal step.	Implemented the "Fractal Clustering" (Iteration 2) on the subset. Computed and analyzed cluster quality metrics (SSE, Silhouette Score). Engineered the final is_golden_cluster binary target variable for the dataset.

Assignment 3 (Week 4-5)	**Supervised Learning**	Led the data amalgamation, performing Reverse Geocoding and Spatial Joins to merge datasets. Conducted the **Ensemble Model Analysis (Random Forest, XGBoost)**, achieving the project's top F1 scores.	Co-led data amalgamation, focusing on Attribute Left Joins. Conducted the **Instance-Based & Boundary Model Analysis (KNN, SVM)**, establishing a performance baseline and comparing model families.
Assignment 4 (Week 6)	**Interactive Dashboard**	Developed and tested the data resampling pipelines (SMOTE, RandomUnderSampler). Analyzed the performance and sensitivity of the **MLP (Neural Network) and SVM** models across different class balances.	Designed and built the interactive dashboard front-end (UI/UX) using ipywidgets. Analyzed the performance of the **XGBoost and Random Forest** models, noting their high robustness to class imbalance.
Assignment 5 (Week 7)	**Feature Optimization**	Calculated feature importance using Random Forest, which identified 'Walkability' as noise. Retrained and evaluated the **MLP and SVM** models on the optimized (4-feature) dataset to confirm performance gains.	Formulated the optimization hypothesis. Retrained and evaluated the high-performing **XGBoost and Random Forest** models. Authored the final analysis on the business impact of improved

			efficiency and accuracy.
Assignment 6 (Week 7)	**Regression Analysis**	Engineered the **'Host Experience Score'** and **'Popularity Score'** latent features. Ran and tuned the Keras (MLP) and other linear regression models to test the new features.	Engineered the **'Space Efficiency Score'** latent feature. Ran and tuned the tree-based regression models (**Random Forest, XGBoost**), identifying the final, best-performing model for price prediction.