Final Project Report:

Antalya Rental Prices Predictions

Submission Date: June 10, 2025
Submitted by: **Kaan Yazıcıoğlu (97364)**

1. About the Data

This dataset was sourced from Kaggle and contains rental apartment listings from the Muratpaşa district of Antalya, Turkey. The original dataset was in Turkish, and I translated all feature names into English for clarity. After cleaning and preprocessing, the dataset included:

- **Total records used:** 712
- **Training set:** 80% (569 records)
- **Testing set:** 20% (143 records)

Target variable: \`rent_price\` (monthly rent in Turkish Lira currency)

Selected input features:

- \`rooms\` ("2+1" converted to 3) (integer)
- \`net_m2\` (usable area in square meters) (integer)
- \`elevator\` (binary: 0 or 1) (bool)
- \`compound\` (binary: 0 or 1) (bool)
- \`fee\` (monthly maintenance cost) (integer)
- \`furnished\` (binary: 0 or 1) (bool)
- \`is_new_building\` (1 = building younger than 15 years, 0 = 15+ years old) (bool)

2. Data Cleaning and Feature Engineering

Several preprocessing steps were required to prepare the dataset:

- Replaced Turkish column headers with English equivalents.
- Converted `rooms` column (e.g., "3+1") to total room count using custom function.
- Converted `building_age` text categories (e.g., "21-25") to approximate numeric values.
- Removed rows with missing or invalid values in `rooms` and `building_age`.
- Created a new binary feature: `is_new_building` = 1 if building is newer than 15 years.
- Removed extreme outlier records where `rent_price > 50000`, which reduced model overfitting and improved generalization.

---

3. Model Development and Comparison

I developed and compared two regression models to predict rental prices:

Linear Regression

- Simple and interpretable model.
- **Train R^2 score:** 0.356
- **Test R^2 score:** 0.303

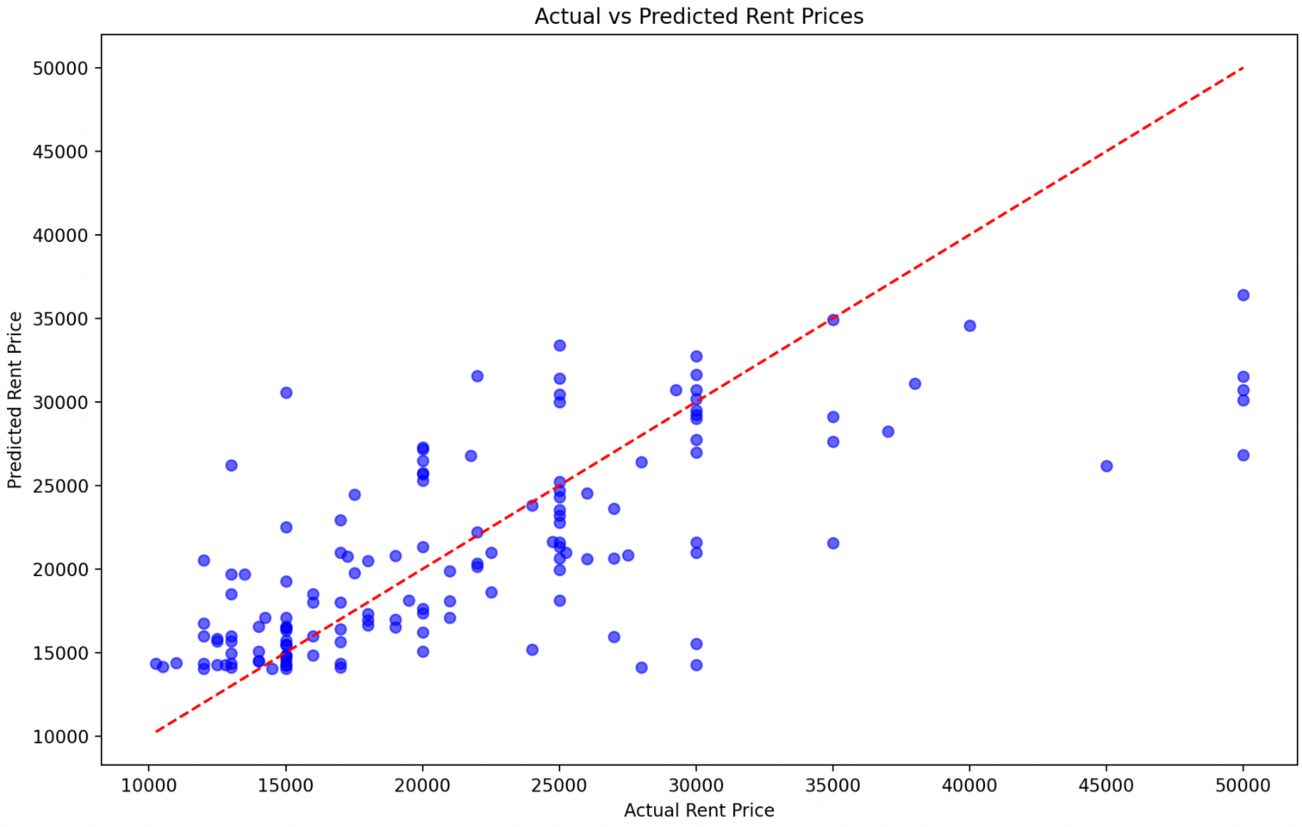
Random Forest Regressor

- Tree-based ensemble model that can learn non-linear patterns.
- **Train R^2 score:** 0.716
- **Test R^2 score:** 0.490

After tuning the input features and removing outliers, Random Forest outperformed Linear Regression in both accuracy and stability.

4.1 Actual vs Predicted Rent Prices

This section shows the comparison between actual rent prices and predicted rent prices from the Random Forest model. The scatter plot below demonstrates how closely the model's predictions align with the actual values. A perfect prediction would result in all points lying on the red dashed line. The closer the points are to this line, the more accurate the model's predictions are.



- **Ideal Prediction**: Points that lie on the red line represent perfect predictions, where the predicted price is exactly equal to the actual price.
- **Model Performance**: Most of the points cluster around the red line, indicating that the model is fairly accurate. However, there are some points scattered away from the line, especially at the higher end of the rent prices (above 30,000 TL), which suggests the model struggles with predicting very high rental values.

4.2 Error Analysis and Model Evaluation

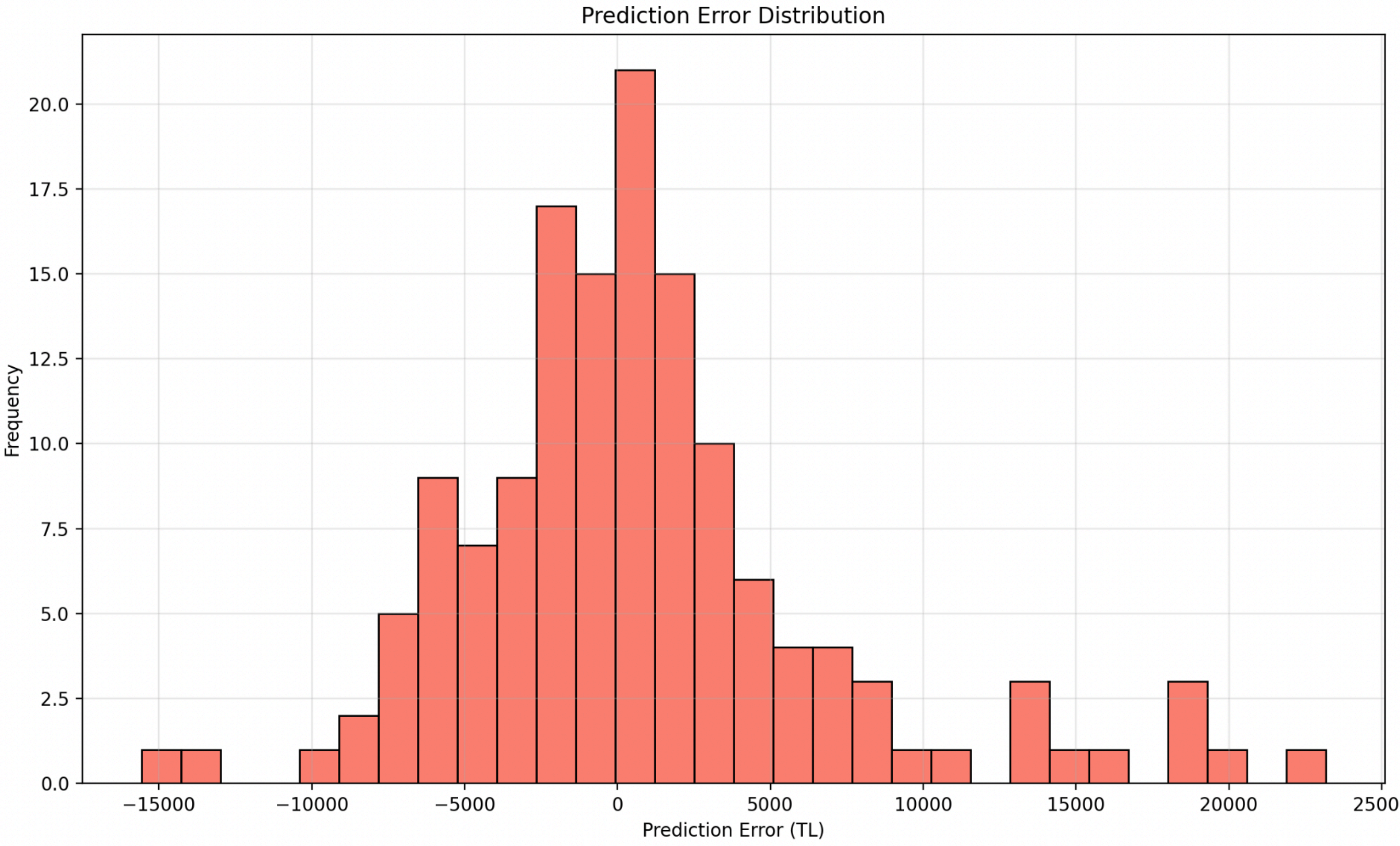
To measure how far off the model predictions were from the actual rent prices:

- **Mean Absolute Error (MAE):** 4307 TL
- **Root Mean Squared Error (RMSE):** 6264 TL

The histogram of prediction errors showed that:

- Most errors fall within the ±5000 TL range.
- Error distribution was fairly symmetric, indicating no major bias.

These results suggest that while the model is not perfect, it provides useful estimates within a realistic margin of error.

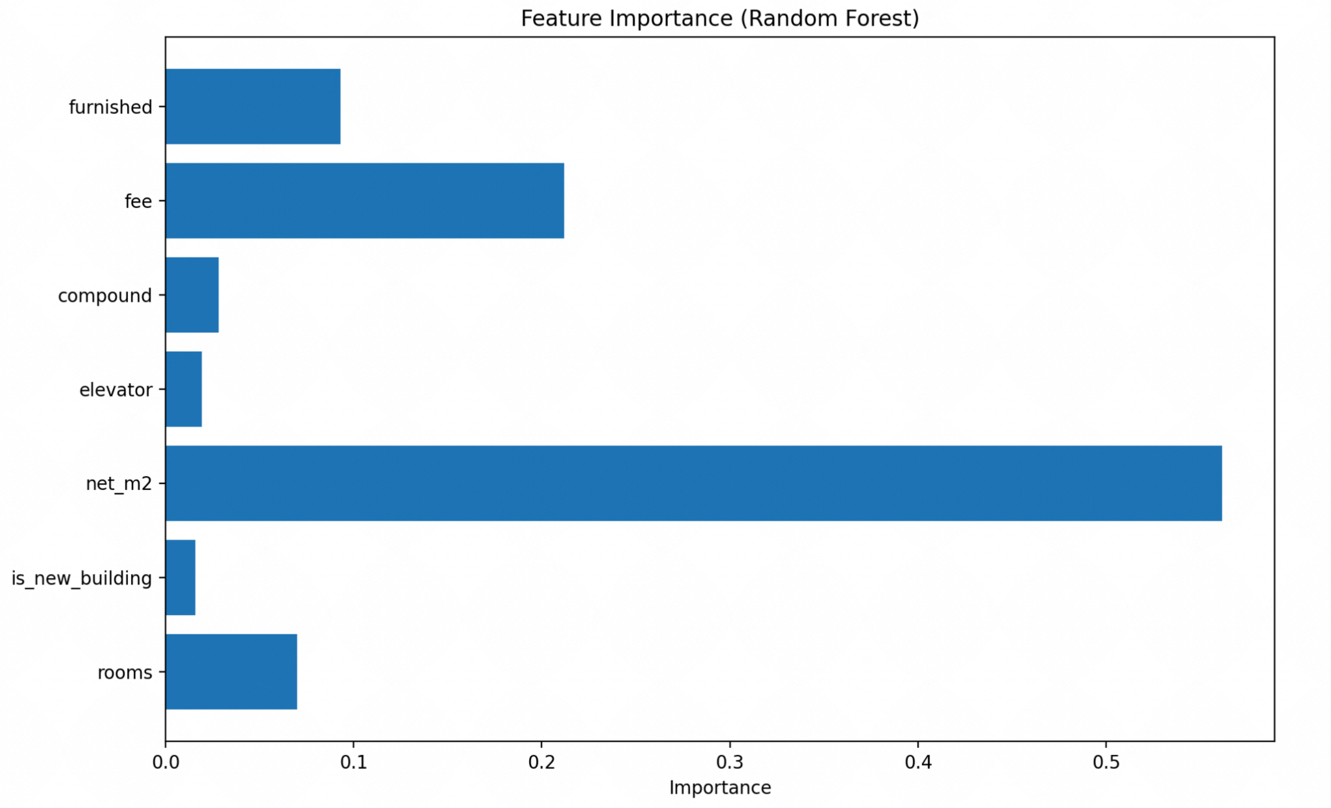


5. Feature Importance

Using the feature importance attribute of the Random Forest model, I ranked the features:

1. `net_m2` – the strongest predictor
2. `fee` – positive correlation with rent
3. `furnished` – minor but noticeable effect
4. `rooms` – contributes moderately
5. `is_new_building` – low but meaningful influence
6. `elevator`, `compound` – minimal impact

This analysis helped validate the choice of input variables and showed that usable area is the most critical driver of rental price.



6. What I Learned

- Net usable area and monthly fees are the most important features in predicting rent.
- Outlier removal significantly improves model performance.
- Tree-based models like Random Forest are better suited for real estate pricing tasks.
- Creating a new binary feature (`is_new_building`) helped simplify and improve predictions.

---

7. Limitations

Although the model works well, several limitations were noted:

Limitations

- The model does not include **geolocation** or neighborhood-level data, which are key factors in rental price variations.
- Features like heating type, parking, or floor level were **not included** in the final model.
- The model does not account for seasonality or long-term market changes.
- The dataset of **712 records** might not be large enough to capture all rental price variations, limiting model generalization.

Improvements for future versions:

- Include geolocation and more property features (heating type, parking, floor level).
- Account for seasonality and market trends to improve predictions.
- Expand the dataset for better generalization.
- Use more advanced models like Gradient Boosting or XGBoost..

8. Summary and Final Thoughts

This project showed how to use real estate data to build a basic rental price prediction model. After preprocessing and experimentation with different algorithms, I achieved an R^2 score of **0.49** on the test set using Random Forest.

While not perfectly accurate, this model offers a solid baseline for automated rental estimation. Further improvements could make it practical for real estate agents or listing platforms.

Overall, this project gave me valuable hands-on experience with:

- Data preprocessing
- Feature engineering
- Regression models
- Model evaluation
- Real-world data challenges