Optimizing a Real Estate Investment Portfolio in Jabodetabek Using Machine Learning and Dynamic Programming Within Budget Constraints

Integrating Predictive Models with Sequential Decision-Making for Residential Property Investmen

Asybel B.P. Sianipar - 15223011
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail: asybel.bintang@gmail.com , 15223011@mahasiswa.itb.ac.id

***Abstract*—This paper presents a two-stage framework for optimizing real estate investment portfolios in Jabodetabek using machine learning and dynamic programming. The first stage forecasts property-level Net Operating Income (NOI) and capital appreciation by training XGBoost models on historical sale and rental data. These predictions are used to compute the Net Present Value (NPV) of each property over a 10-year investment horizon. In the second stage, the portfolio selection problem is formulated as a 0/1 knapsack problem and solved using dynamic programming, with greedy and brute-force methods implemented as benchmarks. Experimental results show that the dynamic programming approach consistently identifies higher-value portfolios under a fixed budget constraint, outperforming heuristic methods in both performance and reliability. This work highlights the benefit of combining predictive analytics with combinatorial optimization to support data-driven decision making in property investment.**

Keywords—dynamic programming; machine learning; real estate; investment;

I. Introduction

Real estate is the largest asset class in the world, encompassing both residential and commercial properties. Its prominence stems from its significant contribution to global wealth and its dual function as both a store of value and an investment vehicle.

The Indonesian residential real estate market presents a unique set of challenges and opportunities. Unlike in many Western countries where transaction data is publicly recorded and accessible, the Indonesian market operates with significant data opacity. This lack of transparent historical pricing forces investors to rely on anedoctal evidence, localized knowledge, and simple heuristics. This challenge is particularly acute in the Jabodetabek (Jakarta, Bogor, Depok, Tangerang, and Bekasi). As the nation's primary economic engine, Jabodetabek is highly dynamic and heterogeneous region, composed of countless micro-markets where property values are driven by complex interplay of infrastructure development, developer activity, and shifting demographic trends.

For real estate investors, the primary objective is to maximize total return, which is a composite of rental income—typically measured as Net Operating Income (NOI)—and long-term capital appreciation. Traditional investment strategies often involve ranking properties by simple metrics like capitalization rate (NOI divided by price). However, such methods favor immediate yield and may fail to identify properties with lower initial returns but significantly higher future growth potential. In a diverse market like Jabodetabek, a strategy that cannot simultaneously evaluate both income and growth potential at a granular, property-by-property level is fundamentally incomplete. This gap highlights the need for holistic, forward-looking valuation and selection framework.

This paper addresses this challenge by developing and evaluating an integrated framework that combines machine learning for forecasting with dynamic programming for optimization. This paper aims to move beyond simple heuristics and provide a method for constructing an optimal investment portfolio under a fixed budget. The central research question is: Can a portfolio strategy, driven by machine learning-based forecasts of NPV and optimized using an exact method like dynamic programming, generate a significantly higher total return thatn portfolios constructed using common greedy heuristics?

Disclaimer: None of the numbers, predictions, or models discussed in this paper should be used as a basis for real investment decisions. This paper is intended solely for educational and experimental purposes and does not constitute financial advice.

II. Literature Study

A. Hedonic Pricing and Machine Learning for Valuation

The foundational model for real estate valuation is the hedonic pricing model, which posits that a property's price is a function of its constituent characteristics. Early research in this area, such as the seminal work by Rosen, applied multiple linear regression to determine the marginal contribution of attributes

like square footage, number of bedrooms, and location to the overall price [1]. This approach has been widely applied to real estate markets globally, including an early model for the Greater Jakarta area by Samapatti and Tay, which highlighted that locational attributes were key drivers of new housing prices [2].

While foundational, traditional regression models often fail to capture the complex, non-linear relationships and feature interactions inherent in real estate data. The last decade has seen a paradigm shift towards the use of machine learning (ML) for property valuation. Numerous studies have demonstrated that ML algorithms—such as Random Forests, Support Vector Machines, and Gradient Boosted Trees—consistently outperform hedonic regressions in predictive accuracy [3].

B. Real Estate Portfolio Optimization

Moving from the valuation of single assets to the selection of an optimal group of assets is the domain of portfolio optimization. The classic framework is Modern Portfolio Theory (MPT), developed by Harry Markowitz, which constructs portfolios based on the trade-off between expected return and risk, measured as variance [4]. However, MPT has significant limitations when applied to real estate. It's assumptions of asset divisibility and liquidity are violated by real estate assets, which are indivisible, unique, and involve high transaction costs.

This has led researchers to explore alternative methods from the field of operations research. Multi-Criteria Decision-Making (MCDM) models, such as the Analytic Hierarchy Process (AHP), have been used to rank properties based on a combination of financial and non-financial factors [5].

C. Literature Study Summary and Research Gap

The literature provides a clear foundation for my work. Machine larning, particularly XGBoost, is the state-of-the-art for property valuation. Various optimization techniques exist for porrtfolio selection, but many are ill-suited for the specific constraints of real-estate investing. Finally, the indonesian market's data opacity necessitates new approaches to fundamental tasks like estimating price growth.

A clear gap emerges from this review: there is no integrated framework that combines property-level forecasting of both NOI and capital appreciation using state-of-the-art ML and then feeds these predictions into an exact optimization algorithm like Dynamic Programming to construct a provably optimal portfolio under a budget constraint. My study directly addresses this gap. This paper move beyond simple heuristics to create a data-driven, end-to-end solution tailored to the specific challenges and opportunities of the Jabodetabek real estate market.

III. Foundational Theory

A. Dynamic Programming

Dynamic Programming (DP) is a powerful algorithmic technique for solving complex problems by breaking them down into a collection of simpler, overlapping subproblems. Developed by Richard Bellman in the 1950s, the approach is predicated on the principle of optimality. The Bellman principle of optimality states that an optimal policy has a property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision.

For a problem to be solvable with DP, it must exhibit two key properties:

- Optimal Substructure: An optimal solution to the overall problem can be constructed from the optimal solutions of it's subproblems.
- Overlapping Subproblems: The algorithm solves the same subproblems repeatedly. DP avoids recomputation by storing the solutions to these subproblems in a table (a technique known as memoization) or by solving them in a systematic, bottom-up fashion.

In this paper, DP is used as the exact solution method for the 0/1 Knapsack Problem to find the provably optimal investment portfolio under the budget constraint. Below is the example of dynamic programming problem.

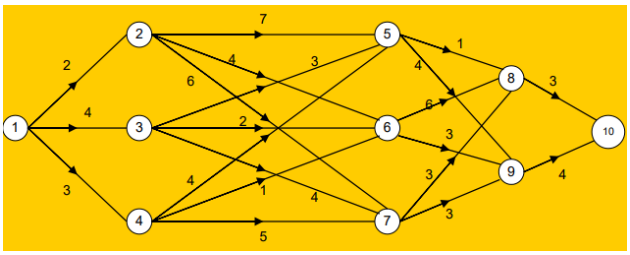


Fig. 1. Dynamic Programming Graph (Source: https://informatika.stei.itb.ac.id/~rinaldi.munir/)

Recurrence relationship of the problem:

- True basis:

$$f_0(s) = 0$$

- First node:

$$f_1(s) = cost(d_0, s)$$

- Recurrence:

$$f_k(s) = \min_{d \in D}\{f_{k-1}(d) + cost(d, s)\}$$

With:

- $f_k(s)$: function to find the minimum cost of a state s in stage k by evaluating previous subproblems.
- D : list of decisions
- s : a state that is being evaluated of the stage k.
- $cost(d, s)$: cost from a node from previous stage to the state that is being evaluated.

B. *The 0/1 Knapsack Problem*

The 0/1 Knapsack Problem is a classic problem in combinatorial optimization. It describes a situation where one must choose from a set of items, each with an associated weight and value, to pack into a knapsack that has a maximum weight capacity. The goal is to maximize the total value of the items in the knapsack without exceeding the capacity.

The ”0/1” is crucial; for each item, the decision is binary—either the item is taken (1) or it is left behind (0). It is not possible to take a fraction of an item. Formally, given n items, the problem is to:

Maximize:

$$\sum_{i=1}^{n} v_i x_i$$

Subject to:

$$\sum_{i=1}^{n} w_i x_i \leq W$$

$$x_i \in \{0,1\}$$

Where:

- V_i is the value of item i
- w_i is the weight of item i
- W is the maximum capacity of the knapsack
- X_i is the decision variable, x_i \in {0, 1}

C. *Extreme Gradient Boosting (XGBoost)*

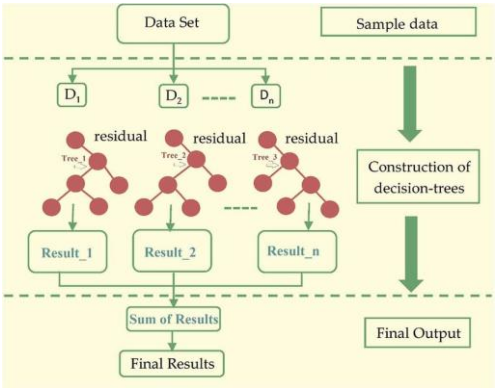


Fig. 2. XGBoost Architecture Diagram (Source: https://www.researchgate.net/figure/General-architecture-of-XGBoost-algorithm_fig2_371285048)

Extreme Gradient Boosting (XGBoost) is a highly efficient and scalable machine learning algorithm based on the gradient boosting framework. Developed by Chen and Guestrin, XGBoost builds a predictive model in the form of an ensemble of weak prediction model, typically decision trees [6].

XGBoost builds on gradient boosting by optimizing a regularized objective function, which balances prediction accuracy with model complexity. The general form is:

$$Obj(\theta) = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k)$$

where l is the loss function (e.g., mean squared error for regression), $\hat{y}_i = \sum_{k=1}^{K} f_k(x_i)$ represents the model prediction as the sum of K regression trees, and $\Omega(f_k)$ penalizes each tree’s complexity via:

$$\Omega(f_k) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2$$

where T is the number of leaves, and w_j are the leaf scores. Training proceeds by additively building trees to minimize this objective.

D. *Net Present Value (NPV)*

Net Present Value (NPV) is a cornerstone of corporate finance used to evaluate the profitability of an investment or project. The principle is based on the time value of money: a sum of money today is worth more than the same sum in the future due to its potential earning capacity. NPV calculates the present-day value of all future cash flows generated by an investment, minus the initial cost of the investment. The foundational theory and rigorous definition of NPV are detailed in authoritative finance texts [7].

The foundational formula for NPV is:

$$NPV = \sum_{t=1}^{T} \frac{CF_t}{(1+r)^t} - C_0$$

where:

- T : investment horizon
- r : discount rate
- CF_t : net cash flow in period t
- C_0 : initial capital

The equation above can be specified for a real estate investment as follows:

$$NPV = \underbrace{\left(\sum_{t=1}^{T} \frac{NOI_i}{(1+r)^t}\right)}_{PV\ of\ Rental\ Income} + \underbrace{\frac{\hat{P}_{i,2025+T}}{(1+r)^T}}_{PV\ of\ Future\ Sale} - \underbrace{P_{i,2025}}_{Initial\ Cost}$$

where:

- T : investment horizon
- NOI : Net Operating Income
- r : discount rate
- $\hat{P}$: predicted future price

- *P* : initial purchase price

IV. Methodology

This study proposes a two-stage framework for real estate investment portfolio optimization. The first stage uses machine learning models to forecast property valuations, and the second stage employs dynamic programming to select an optimal portfolio under a fixed budget constraint.

All modeling, data handling, and optimization tasks were implemented in **Python**, leveraging libraries such as pandas, scikit-learn, xgboost.

A. *Data Acquisition and Pre-processing*

The datasets used for this study are:

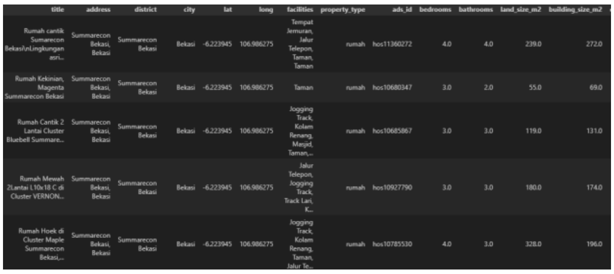
- Residential Property in Jabodetabek Sales Price 2023 (from Kaggle)

  https://www.kaggle.com/datasets/nafisbarizki/daftar-harga-rumah-jabodetabek

- Residential Property Sales Price 2025 in Indonesia (scraped from Rumah123.com)

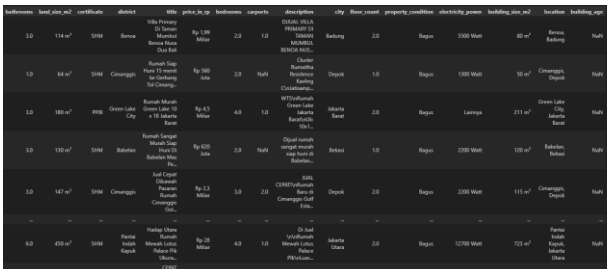
- Residential Property Rental 2025 in Indonesia (scraped from Rumah123.com)

Data scraping was conducted using repectful and compliant methods that follow robots.txt of Rumah123.com and Rumah123's terms of service. Key features extracted: title, district, city, price, bedrooms, bathrooms, land_area, buidling_area, location, certificate, electricity_power, building_condition, and description.



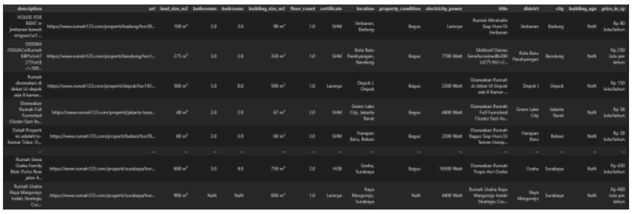
title	address	district	city	lat	long	facilities	property_type	ads_id	bedrooms	bathrooms	land_size_m2	building_size_m2
Rumah cantik Sumarecon Bekasi\nLingkungan asri...	Summarecon Bekasi, Bekasi	Summarecon Bekasi	Bekasi	-6.223945	106.986275	Tempat Jemuran, Jalur Telepon, Taman, Taman	rumah	hos11360272	4.0	4.0	239.0	272.0
Rumah Kekinian, Magenta Summarecon Bekasi	Summarecon Bekasi, Bekasi	Summarecon Bekasi	Bekasi	-6.223945	106.986275	Taman	rumah	hos10680347	3.0	2.0	55.0	69.0
Rumah Cantik 2 Lantai Cluster Bluebell Summare...	Summarecon Bekasi, Bekasi	Summarecon Bekasi	Bekasi	-6.223945	106.986275	Jogging Track, Kolam Renang, Masjid, Taman,...	rumah	hos10685867	3.0	3.0	119.0	131.0
Rumah Mewah 2Lantai L10x18 C di Cluster VERNON...	Summarecon Bekasi, Bekasi	Summarecon Bekasi	Bekasi	-6.223945	106.986275	Jalur Telepon, Jogging Track, Track Lari, K...	rumah	hos10927790	3.0	3.0	180.0	174.0
Rumah Hoek di Cluster Maple Summarecon Bekasi,...	Summarecon Bekasi, Bekasi	Summarecon Bekasi	Bekasi	-6.223945	106.986275	Jogging Track, Kolam Renang, Taman, Jalur Te...	rumah	hos10785530	4.0	3.0	328.0	196.0

Fig. 3. 2023 Sale data preview prior to cleaning



bathrooms	land_size_m2	certificate	district	title	price_in_rp	bedrooms	carports	description	city	floor_count	property_condition	electricity_power	building_size_m2	location	building_age
3.0	114 m^2	SHM	Benoa	Villa Primary Di Taman Mumbul Benoa Nusa Dua Bali	Rp 1,99 Miliar	2.0	1.0	DIJUAL VILLA PRIMARY DI TAMAN MUMBUL BENOA NUS...	Badung	2.0	Bagus	5500 Watt	80 m^2	Benoa, Badung	NaN
1.0	64 m^2	SHM	Cimanggis	Rumah Siap Huni 15 menit ke Gerbang Tol Cimang...	Rp 560 Juta	2.0	NaN	Cluster Rumaitha Residence Kavling C\nJatisamp...	Depok	1.0	Bagus	1300 Watt	50 m^2	Cimanggis, Depok	NaN
3.0	180 m^2	PPJB	Green Lake City	Rumah Murah Green Lake 10 x 18 Jakarta Barat	Rp 4,5 Miliar	4.0	1.0	WTS\nRumah Green Lake Jakarta Barat\nUk 10x1...	Jakarta Barat	2.0	Bagus	Lainnya	211 m^2	Green Lake City, Jakarta Barat	NaN
3.0	130 m^2	SHM	Babelan	Rumah Sangat Murah Siap Huni Di Babelan Mas Pe...	Rp 620 Juta	2.0	NaN	Dijual rumah sangat murah siap huni di Babelan...	Bekasi	1.0	Bagus	2200 Watt	120 m^2	Babelan, Bekasi	NaN
3.0	147 m^2	SHM	Cimanggis	Jual Cepat Dibawah Pasaran Rumah Cimanggis Gol...	Rp 2,3 Miliar	3.0	2.0	JUAL CEPAT\nRumah Baru di Cimanggis Golf Esta...	Depok	2.0	Bagus	2200 Watt	115 m^2	Cimanggis, Depok	NaN
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
6.0	450 m^2	SHM	Pantai Indah Kapuk	Hadap Utara Rumah Mewah Lotus Palace Pik Ukura...	Rp 28 Miliar	4.0	1.0	Di Jual \n\nRumah Mewah Lotus Palace Pik\nLuas...	Jakarta Utara	2.0	Bagus	12700 Watt	723 m^2	Pantai Indah Kapuk, Jakarta Utara	NaN

Fig. 4. 2025 Sale data preview prior to cleaning



description	url	land_size_m2	bathrooms	bedrooms	building_size_m2	floor_count	certificate	location	property_condition	electricity_power	title	district	city	building_age	price_in_rp
HOUSE FOR RENT in jimbaran bawah imigrasi\n3 ...	https://www.rumah123.com/properti/badung/hos38...	100 m^2	2.0	3.0	90 m^2	1.0	SHM	Jimbaran, Badung	Bagus	Lainnya	Rumah Minimalis Siap Huni Di Jimbaran	Jimbaran	Badung	NaN	Rp 80 Juta/tahun
[illegible]	https://www.rumah123.com/properti/bandung/hos1...	275 m^2	3.0	2.0	330 m^2	1.0	SHM	Kota Baru Parahyangan, Bandung	Bagus	7700 Watt	[illegible]	Kota Baru Parahyangan	Bandung	NaN	Rp 250 Juta per tahun
Rumah disewakan di dekat UI depok ada 8 kamar...	https://www.rumah123.com/properti/depok/hos182...	500 m^2	5.0	8.0	500 m^2	1.0	Lainnya	Depok I, Depok	Bagus	2200 Watt	Disewakan Rumah di dekat UI Depok ada 8 Kamar ...	Depok I	Depok	NaN	Rp 150 Juta/tahun
Disewakan Rumah Full Furnished Cluster East As...	https://www.rumah123.com/properti/jakarta-bara...	48 m^2	2.0	2.0	67 m^2	2.0	SHM	Green Lake City, Jakarta Barat	Bagus	4400 Watt	Disewakan Rumah Full Furnished Cluster East As...	Green Lake City	Jakarta Barat	NaN	Rp 58 Juta/tahun
[illegible]	https://www.rumah123.com/properti/bekasi/hos39...	60 m^2	2.0	3.0	60 m^2	2.0	SHM	Harapan Baru, Bekasi	Bagus	2200 Watt	Disewakan Rumah Bagus Siap Huni Di Taman Harap...	Harapan Baru	Bekasi	NaN	Rp 28 Juta/tahun
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
[illegible]	https://www.rumah123.com/properti/surabaya/hos...	600 m^2	3.0	4.0	750 m^2	2.0	HGB	Graha, Surabaya	Bagus	16500 Watt	Disewakan Rumah Tropis Asri Graha	Graha	Surabaya	NaN	Rp 430 Juta/tahun
Rumah Usaha Raya Margorejo Indah Strategis Coc...	https://www.rumah123.com/properti/surabaya/hos...	900 m^2	NaN	NaN	600 m^2	1.0	Lainnya	Raya Margorejo, Surabaya	NaN	4400 Watt	Rumah Usaha Raya Margorejo Indah Strategis Coc...	Raya Margorejo	Surabaya	NaN	Rp 400 Juta per tahun

Fig. 5. 2025 Rent data preview prior to cleaning

The preprocessing steps were designed to ensure data quality, consistency, and compatibility across datasets. The following steps were applied, as detailed in the data preparation pipeline:

1. Column Cleanup and Reordering:
2. Aligning Historical Data
3. Data Type Conversion
4. Categorical Data Cleaning
5. Geographic Filtering
6. Duplicate and Null Value Handling
7. Outlier Removal
8. Feature Encoding and Standardization

These preprocessing steps resulted in robust, consistent datasets suitable for machine learning and portfolio optimization:

- df_sale_2023
- df_rent
- df_sale_2025_train (for training market value prediction in 2025)
- df_sale_2025 (for final test + optimization)

B. *Exploratory Data Analysis*

To understand the structure and characteristics of the residential property dataset, we conducted an Exploratory Data Analysis (EDA) on 6,227 cleaned listings collected from Jabodetabek in 2025. This step is essential for uncovering patterns, detecting anomalies, and forming hypotheses that can guide downstream modeling and portfolio selection.

The dataset comprises 14 features for each property, including pricing, physical attributes (e.g., land size, number of bedrooms), categorical variables (e.g., location, certificate type), and textual descriptions.

Key numeric variables include:

- price_in_rp: Listed property price in Indonesian Rupiah
- land_size_m2, building_size_m2: Physical dimensions
- bedrooms, bathrooms, floor_count: Functional capacity
- electricity_power: Installed capacity in VA

Categorical fields include:

- city, district: Geographic location

- certificate, property_condition: Legal and physical attributes

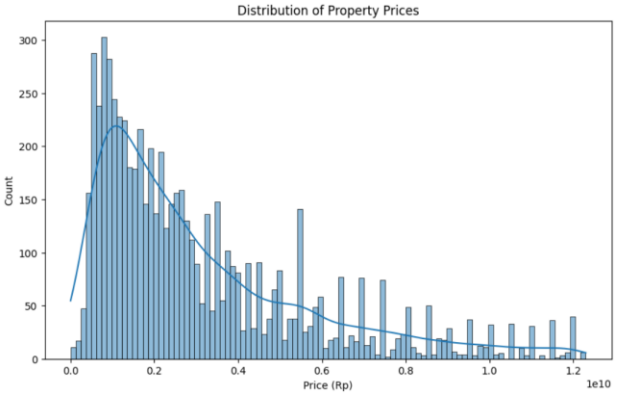


Fig. 6. Distribution of Jabodetabek property prices (Source: Author's archive)

The distribution of property prices (in Indonesian Rupiah) is visualized to assess market spread and skewness. The histogram above shows a right-skewed distribution of property prices, indicating that most listings fall within the lower-to-mid price range, with fewer high-end properties. This long tail suggests the presence of a luxury segment with significantly higher prices.

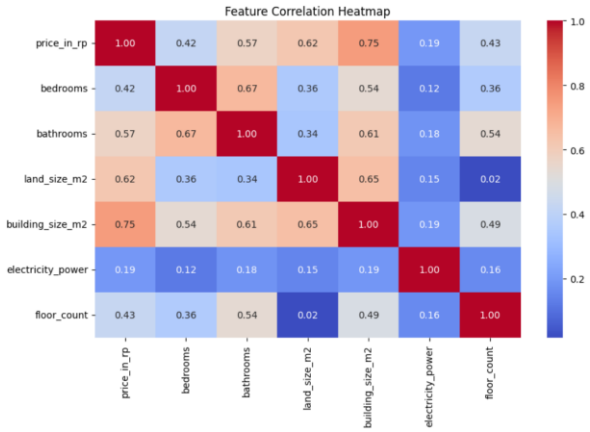


Fig. 7. Jabodetabek property features correlation heatmap (Source: Author's archive)

A heatmap of the Pearson correlation coefficients between numeric features was generated to understand linear relationships. Features included: price_in_rp, bedrooms, bathrooms, land_size_m2, building_size_m2, electricity_power, floor_count.

This matrix highlights strong positive correlations between:

- Land and building size
- Bedrooms and building size
- size and building size to Price

Fig. 8. Top 10 Jabodetabek cities by median property price (Source: Author's archive)

Finally, we analyzed price variations across geographic regions. Figure 8 shows the top 10 cities by median property price:

Observations:

- Jakarta Utara (North Jakarta), Jakarta Selatan (South Jakarta), and Jakarta Barat (West Jakarta) dominate in terms of median pricing.
- Tangerang and Jakarta Timur follow as secondary premium zones.
- Cities like Bekasi, Bogor, and Depok exhibit significantly lower median prices, making them attractive for budget-conscious investors or first-time buyers.

This geographic breakdown highlights regional disparities in valuation and potential for location-based portfolio diversification.

C. Asset Value Forecasting

The forecasting phase employs XGBoost to predict property sale prices and rental income. This is critical for estimating financial returns. Three models were developed to provide comprehensive projections, leveraging preprocessed datasets to ensure accuracy. The process involves feature preparation, log-transformation of targets, hyperparameter tuning, and financial metric calculations.

Model Development:

- 2023 Sale Model: Trained on df_sale_2023 to predict historical sale prices (price_in_rp).
- 2025 Rent Model: Trained on df_rent_2025 to predict annual rental income (rent_price_per_year).
- 2025 Sale Model: Trained on df_sale_2025_train (50% split from df_sale) to predict 2025 sale prices.

Feature Preparation:

- Dropped non-predictive columns (url, title, description, price_in_rp or rent_price_per_year)

- One-hot encoded categorical features with drop-first=True, aligning 2023 and 2025 datasets using DataFrame.reindex.
- Standardized numerical features for cross-temporal comparability.
- np.log1p() (a method from numpy) is used to targets to address price skewness. The predictions can be reverted using np.expm1().

After these preprocessing steps, the data was ready for model training. Below is the code snapshot of the XGBoost regressor model setup.

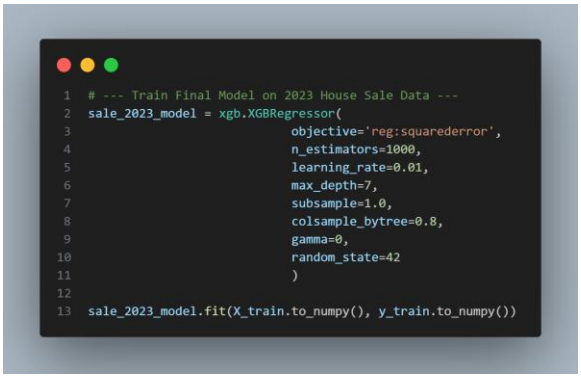


Fig. 9. XGBoost Model Code Snapshot (Source: Author's archive)

Training Configuration (optimized hyperparameters via GridSearchCV [3-fold CV, negative MSE scoring):

TABLE I. XGBOOST HYPERPARAMETERS

n_estimators	1000
learning_rate	0.01
max_depth	7
subsample	1.0
colsample_bytree	0.8
gamma	0

Using these models, the df_sale_2025 properties were enriched with these financial metrics:

- NOI Predictions (from rent model)

$$NOI_i = Rent_i - NOE$$

Where NOE is Net Operating Expenses. We assume that 30% of the gross rental income is allocated to NOE, which include property taxes, maintenance, management fees, and vacancy allowances. This 70/30 split between Net Operating Income (NOI) and expenses is a common benchmark in property valuation models, especially for residential assets in emerging markets.

- Capital Gain Forecast via CAGR (Compound Annual Growth Rate):

$$CAGR_i = \sqrt{\left(\frac{\hat{P}_{i,2025}}{\hat{P}_{i,2023}}\right)} - 1$$

- NPV Calculation

$$NPV_i = \sum_{t=1}^{10} \frac{NOI_{i,t}}{(1.1)^t} + \frac{\hat{P}_{i,2035}}{(1.1)^{10}} - P_{i,2025}$$

Where:

  - NOI grows by 3% annually
  - $\hat{P}_{i,2035} = \hat{P}_{i,2025} \times (1 + CAGR)^{10}$

The calculation assumes a discount rate (r) of 10% and a conservative annual rental growth rate of 3%. This NPV serves as the objective value for optimization.

A discount rate of 10% is used to reflect the opportunity cost of capital and the perceived risk of investing in real estate within the Jabodetabek region. This rate captures market uncertainties, inflation expectations, and the investor's required rate of return. It is also consistent with published rates used in feasibility studies and REIT evaluations in Southeast Asia.

The model assumes a modest 3% annual growth in rent, which aligns with the long-term average rental inflation in Indonesia's urban residential market. This conservative estimate avoids overstating returns, especially in a market subject to cyclical fluctuations and regulatory constraints.

D. Portfolio Optimization

The Investment selection problem is modeled as a 0/1 Knapsack Problem:

$$max \sum_{i=1}^{n} v_i x_i \quad s.t. \quad \sum_{i=1}^{n} w_i x_i \leq W, \quad x_i \in \{0,1\}$$

Where:

- $v_i = NPV_i$
- $v_i = price_i$
- $W = total\ budget\ (e.g., Rp\ 5\ Billion)$

Dynamic Programming is used to solve this problem exactly by discretizing the budget and storing subproblems values. Algorithms used for benchmarking include:

- solve_portfolio_DP(): optimal solution
- solve_portfolio_greedy(): heuristic rank by NPV
- solve_portfolio_bruteforce(): exhaustive enumeration (used only for small samples due to compute limitations)

The pseudocode for each algorithm is provided below.

```
FUNCTION solve_portfolio_DP(properties, totalBudget)
    SET baseUnit ← 1 million
    SET capacity ← totalBudget ÷ baseUnit
    SET n ← SIZE(properties)

    FOR each property i IN properties DO
        weight[i] ← price[i] ÷ baseUnit
        value[i] ← NPV[i]
    END FOR

    INITIALIZE dp[0…capacity] ← 0
    INITIALIZE keep[0…n][0…capacity] ← FALSE

    FOR i FROM 0 TO n - 1 DO
        FOR w FROM capacity DOWNTO weight[i] DO
            IF dp[w - weight[i]] + value[i] > dp[w] THEN
                dp[w] ← dp[w - weight[i]] + value[i]
                keep[i][w] ← TRUE
            END IF
        END FOR
    END FOR

    SET selected ← EMPTY LIST
    SET w ← capacity

    FOR i FROM n - 1 DOWNTO 0 DO
        IF keep[i][w] THEN
            selected.ADD(i)
            w ← w - weight[i]
        END IF
    END FOR

    RETURN (dp[capacity], selected, SUM(price[i] FOR i IN
selected))
END FUNCTION
```

```
FUNCTION solve_portfolio_greedy(properties, totalBudget)
    FOR each property i IN properties DO
        ratio[i] ← NPV[i] ÷ price[i]
    END FOR

    SORT properties BY ratio DESCENDING

    SET selected ← EMPTY LIST
    SET totalCost ← 0
    SET totalNPV ← 0

    FOR each property i IN sorted properties DO
        IF totalCost + price[i] ≤ totalBudget THEN
            selected.ADD(i)
            totalCost ← totalCost + price[i]
            totalNPV ← totalNPV + NPV[i]
        END IF
    END FOR
```

```
FUNCTION solve_portfolio_bruteforce(properties, totalBudget)
    SET bestNPV ← 0
    SET bestSubset ← EMPTY LIST

    FOR each subset IN ALL_COMBINATIONS(properties) DO
        cost ← SUM(price of each item in subset)
        IF cost ≤ totalBudget THEN
            npv ← SUM(NPV of each item in subset)
            IF npv > bestNPV THEN
                bestNPV ← npv
                bestSubset ← subset
            END IF
        END IF
    END FOR

    totalCost ← SUM(price of items in bestSubset)
    RETURN (bestNPV, bestSubset, totalCost)
END FUNCTION
```

E. Evaluation and Algorithm Comparison

The evaluation phase assesses the performance of the optimization algorithms, comparing their ability to maximize NPV, adhere to the budget, and execute efficiently. Two approaches—static benchmarking and Greedy accuracy assessment—provide insights into algorithmic trade-offs, using the final dataset, rumah123_2025_sale_modelled.csv.

- Static Benchmarking:
  - Dataset: 20-property sample.
  - Metrics: Total NPV, total cost, number of selected properties, and execution time.
  - Procedure: applied each algorithm, summarizing results in a pandas DataFrame.
  - Output: identified the best algorithm by highest NPV.

- Greedy Accuracy Assessment:
  - Dataset: 1000 trials, each sampling 20 properties with random seeds.
  - Metrics: greedy failure rate (greedy's NPV < DP's NPV) and success rate (greedy's NPV = DP's NPV).
  - Procedure: compared greedy and DP solutions per trial. This reports suboptimality frequency.
  - Purpose: evaluated dynamic programming's reliability for large scale applications.

V. Result and Analysis

A. Model Performance

Fig. 10. Model evaluation using RMSE for Sale 2023 Model code snapshot (Source: Author's archive)

TABLE II. RMSE Evaluation for Each Model

	2023 Sale Model	**2025 Rent Model**	**2025 Rent Model**
RMSE (log-scale)	0.26	0.35	0.33
RMSE (IDR)	561,034,167	51,539,800	1,080,427,856

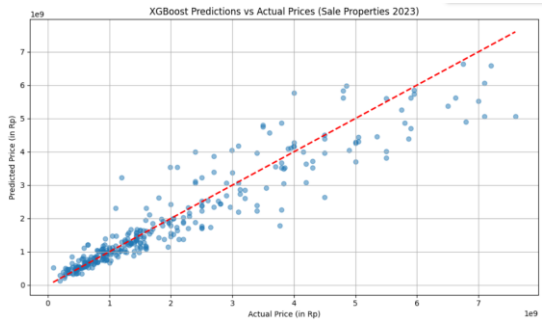


Fig. 11. 2023 Sale Model Test Visualization (Source: Author's archive)

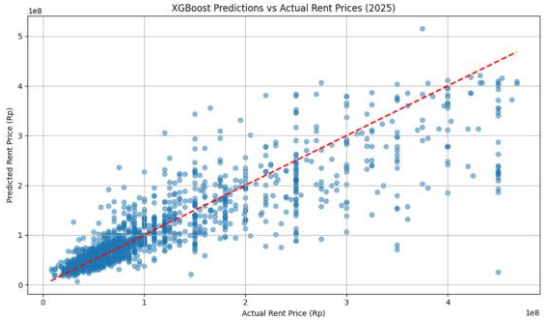


Fig. 12. 2025 Rent Model Test Visualization (Source: Author's archive)

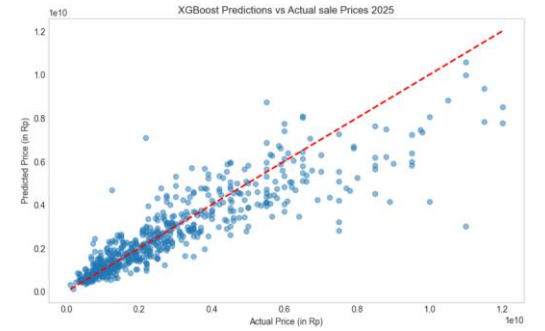


Fig. 13. 2025 Sale Model Test Visualization (Source: Author's archive)

In all three figures, the XGBoost model generally learns the price/rent trends well, but prediction accuracy decreases for high-value properties or rentals. This is reflected in the increasing scatter and deviation from the 1:1 line at higher price ranges.

This could be due to:

- Fewer training samples in the high-price segment (data imbalance),
- More complex or non-linear price drivers for expensive properties (e.g., location prestige, amenities),
- Data noise or inconsistencies in listings for luxury properties.

B. Optimization Algorithm Performance

The first five rows of the final dataset used to evaluate optimization algorithms performace is given below.

price_in_rp	district	city	market_price_2025	market_price_2023	rent_price_per_year_2025	NOI	cap_rate	property_value_growth	NPV
7.000000e+09	Puncak	Bogor	8.386281e+09	1.838574e+09	202866256.0	142006384.0	2.028663	1.135719	5.321842e+12
7.900000e+09	Harjamukti	Depok	8.527677e+09	2.590141e+09	271594016.0	190115808.0	2.406529	0.814486	1.170683e+12
6.900000e+09	Pondok Indah	Jakarta Selatan	7.097878e+09	2.227337e+09	179756960.0	125829872.0	1.823621	0.785136	8.675702e+11
6.760000e+09	Bendungan Hilir	Jakarta Pusat	3.710893e+09	1.177105e+09	110358968.0	77251280.0	1.142770	0.775545	8.049638e+11
1.100000e+10	Pantai Indah Kapuk	Jakarta Utara	5.902377e+09	2.120849e+09	201751472.0	141226032.0	1.283873	0.668240	6.972785e+11

Fig. 14. Final dataset with NPV (Source: Author's archive)

```
Total Properties Available: 1135
Total Budget: Rp 5,000,000,000
============================================================

Comparison of Algorithms on 20-sample:
          Algorithm    Total NPV   Total Cost  Num Items  Time (s) Error
Dynamic Programming 1.861711e+10 4763000000.0          4    0.0138  None
             Greedy 1.859748e+10 4433000000.0          4    0.0010  None
        Brute Force 1.861711e+10 4763000000.0          4    0.8406  None

Greedy vs DP Accuracy over 1000 trials:
 suboptimal 329 times (32.90% failure rate)
Greedy matched DP 671 times (67.10% success rate)

Best Algorithm on 20-sample: Dynamic Programming (NPV = Rp 18,617,112,214)

Details of Selected Properties by Best Algorithm on 20-sample:
      price_in_rp           NPV
2671  2.000000e+09  6.076393e+09
927   7.650000e+08  1.029878e+09
5546  1.000000e+09  3.360650e+08
3607  9.980000e+08  1.117478e+10
```

Fig. 15. Portfolio optimization result comparison for different algorithms (DP, Greedy, and Brute Force) (Source: Author's archive)

To evaluate the effectiveness of different portfolio selection methods, we implemented and tested three algorithms: Dynamic Programming (DP), Greedy Heuristic, and Brute Force Exhaustive Search. Each algorithm was applied to a fixed sample of 20 properties drawn from the final dataset, with a budget cap of Rp 5,000,000,000.

Figure 12 summarizes the performance of the three algorithms. All three were able to select four properties within the budget constraint. However, small differences emerged in the achieved Net Present Value (NPV):

- Dynamic Programming achieved the highest NPV of Rp 18.617 billion.
- Brute Force, as expected, matched DP exactly, confirming its role as a ground truth benchmark.
- Greedy Algorithm, while faster, yielded a slightly lower NPV of Rp 18.597 billion—indicating a ~Rp 20 million shortfall.

Despite the marginal gap, this difference illustrates that even in small samples, heuristic methods can produce suboptimal outcomes, especially when item value-to-cost ratios are not uniformly distributed.

In terms of runtime:

- Greedy was the fastest (~0.0015 seconds),
- followed by Dynamic Programming (~0.0132 seconds),
- and Brute Force (~0.7374 seconds), which becomes intractable for larger inputs.

To further quantify the reliability of each algorithm, we conducted 1000 randomized trials comparing DP and Greedy on 20-property subsamples. The results show:

- Greedy failed to match the DP-optimal result 359 times, a 35.90% failure rate.
- It succeeded in 641 cases (64.10% success rate).

This consistent performance gap validates the necessity of exact methods like Dynamic Programming for high-stakes investment decision-making, especially when the portfolio size and stakes grow larger.

The optimal property selection made by DP in the 200-sample test includes four assets with diverse price points and NPVs (see bottom of Fig. 12), ranging from Rp 765 million to Rp 2 billion. This confirms that the DP-based selection is capable of combining both high-value and cost-effective properties to achieve maximum return under budget constraints.

C. *Portfolio Investment Analysis*

To evaluate the long-term performance of property selection strategies, we simulate a 10-year investment horizon for portfolios chosen using both Dynamic Programming and Greedy algorithms. Each portfolio's projected value incorporates both rental income and capital appreciation, reflecting real-world investment dynamics.

Key components of the simulation:

- Rental Income Stream: Annual Net Operating Income (NOI) is projected to grow at a fixed 3% yearly rate, accounting for rental increases over time.
- Capital Appreciation: Property resale values are modeled using each property's estimated Compound Annual Growth Rate (CAGR), with additional Gaussian noise (σ = 2%) to capture market uncertainty and variance.
- Cumulative Portfolio Value: At each time step, the cumulative value consists of:
    - The discounted NOI cash flows, and
    - The projected resale value of all properties held in the portfolio.
- Comparative Visualization

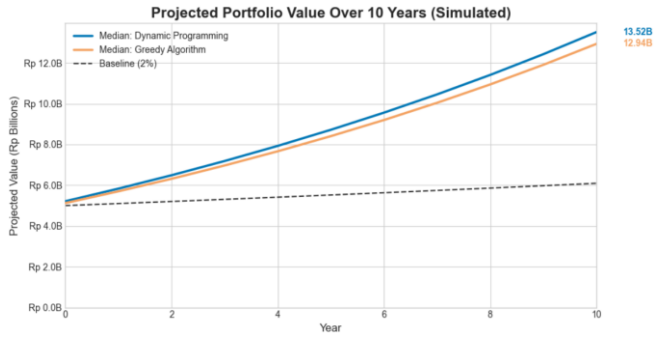


Fig. 16. Simulation 10-year cummulative projection of DP vs Greedy portfolio value (Source: Author's Archive)

This analysis shows how small differences in asset selection—especially early on—can lead to significant compounding effects over time, which can be crucial for long-term investors.

D. *Performance and Algorithm Complexity Analysis*

Our empirical results clearly show that Dynamic Programming (DP) consistently achieves the highest total NPV across repeated trials. In a series of 1,000 randomized tests, DP outperformed the Greedy method 35.9% of the time, Demonstrating that Greedy heuristics often fail to reach optimal solutions when returns-to-cost ratios vary significantly between properties. While the Brute Force approach did match DP's results, it incurred prohibitive computational costs and is infeasible beyond very small portfolio sizes. Time complexity overview:

- DP: $O(nW)$ where n is the number of properties and W is the discretized budget capacity.
- Greedy: $O(n \log n)$ by the sorting step, $O(n)$ for the cost selection.
- Brute Force: $O(2^n)$ due to evaluation of all possible subsets.

TABLE III. ALGORITHM SUMMARY

Algorithm	Time Complexity	Space Complexity	Optimality
DP	$O(nW)$	$O(W)$	Guaranteed Optimal
Greedy	$O(n \log n) + O(n)$	$O(n)$	Approximate (~64% success to match DP)
Brute Force	$O(2^n)$	$O(n)$	Guaranteed optimal

This validates the trade-off: while Greedy is faster, DP ensures optimality within practical time limits for typical real estate portfolio sizes.

VI. CONCLUSION AND SUGGESTIONS

This study presented an integrated, data-driven framework for optimizing residential real estate investment portfolios in the Jabodetabek area by combining machine learning and dynamic programming. The key contribution lies in bridging the gap between predictive modeling and exact optimization under realistic constraints. By forecasting Net Operating Income (NOI) and capital appreciation using XGBoost, and using these projections to compute property-level Net Present Value (NPV), we formulated the portfolio selection task as a 0/1 Knapsack Problem solvable via dynamic programming.

The empirical evaluation demonstrated that:

- The Dynamic Programming (DP) approach consistently achieved the highest total NPV in comparison to heuristic methods like Greedy.

- In randomized trials, Greedy failed to match DP in approximately 36% of the cases, validating the importance of exact optimization when precision is required.
- The simulated growth projections showed the long-term trajectory of selected portfolios and further emphasized the differences in property quality that may not be evident from simple heuristics alone.

Despite these contributions, there are several opportunities for improvement and future work:

- **Rigorous Validation of Assumptions**: While the current study uses conservative financial assumptions—30% operating expense ratio, 10% discount rate, and 3% annual rent growth—it is essential to subject these to deeper empirical scrutiny.
- **Enriching the Feature Set:** Although the existing model explains much of the variability in sale prices, incorporating additional property-level features could enhance predictive performance. Features such as road width, garage/carport availability, dwelling furnishing status, proximity to transit, or micro-neighborhood design (e.g., cul-de-sac vs grid streets) have demonstrated measurable impacts on property value and
- **Stochastic & Risk-Aware Optimization**: Real estate returns are inherently uncertain. Enhancing the optimization framework to incorporate risk-adjusted objectives—including variance, downside risk, or scenario-based outcomes—could improve decision-making under uncertainty. Approaches might include Monte Carlo simulation, robust optimization, or CAPM-calibrated discount rate adjustments.

VII. Attachments

The complete source code, dataset samples, and experimental results used in this paper can be accessed on the project's GitHub repository:

https://github.com/KalengBalsem/RE_Portfolio_Optimization

Acknowledgment

The author expresses sincere gratitude to all parties who have contributed to the completion of this paper, especially to:

- The Almighty God, for His blessings, strength, and guidance throughout the process of conducting and completing this research.
- Dr. Nur Ulfa Maulidev , Dr. Rinaldi Munir, Menterico Adrian, S.T, M.T, lecturers of the IF2211 Algorithm Strategies course, for their invaluable guidance and the knowledge imparted during the lectures.
- The author's parents, for their unwavering moral, emotional, and financial support.

May the kindness and support extended to the author be rewarded in abundance. It is the author's hope that this paper can serve as a meaningful academic contribution for future study and practical application.

References

[1] S. Rosen, "Hedonic prices and implicit markets: product differentiation in pure competition," *Journal of Political Economy*, vol. 82, no. 1, pp. 34–55, 1974.

[2] Samapatti, Supardi, and Linda Tay. "An Hedonic Price Model of New Housing in Indonesia." *Pacific Rim Property Research Journal*, vol. 8, no. 3, 2002, pp. 203–211. doi:10.1080/14445921.2002.11104123.

[3] E. A. Antipov and E. B. Pokryshevskaya, "Mass appraisal of residential apartments: An application of random forest for valuation and a CART-based approach for model diagnostics," Expert Systems with Applications, vol. 39, no. 2, pp. 1772–1778, 2012.

[4] H. M. Markowitz, "Portfolio selection," The Journal of Finance, vol. 7, no. 1, pp. 77–91, 1952.

[5] M. Z. Al-Harbi, "Application of the AHP in project management," *International Journal of Project Management*, vol. 19, no. 1, pp. 19-27, 2001.

[6] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, San Francisco, CA, USA, Aug. 13–17, 2016, pp. 785–794. doi:10.1145/2939672.2939785

[7] R. A. Brealey, S. C. Myers, F. Allen, and A. Edmans, Principles of Corporate Finance, 14th ed. New York: McGraw-Hill, 2022.

[8] R. Munir, *Strategi Algoritma 2024/2025*. [Online]. Available: https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2024-2025/stima24-25.htm

Statement

I hereby declare that the paper I have written is my own work, not an adaptation or a translation of someone else's paper, and not a plagiarism.

Bandung, 24th June 2025

Asybel B.P. Sianipar
15223011