A Nowcasting Approach to Singapore's Property Price Index

1 Issues

The task of forecasting Singapore's property price index is fraught with challenges due to the intricate relationship between various factors, including resale indices and economic indicators. The lag in data availability, especially pronounced in the private sector, makes timely and accurate predictions difficult. Our study, while addressing the general property market, emphasizes the private sector due to its distinct dynamics governed largely by market forces, offering a purer study of economic responses.

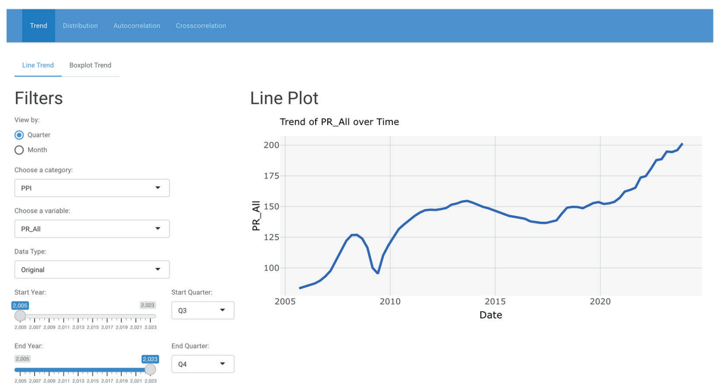
2 Motivation

Our motivation centers on refining forecasts for Singapore's private residential property price index, recognizing its critical role in shaping economic choices. The distinct nature of the private housing market—marked by reduced government intervention—offers a clearer lens through which to view market dynamics. This makes it an essential focus for accurate analysis. Our goal with these precise forecasts is to facilitate informed decision-making for a wide range of stakeholders, including policymakers, investors, real estate professionals, and homeowners. This will enable timely policy adjustments, smarter investment strategies, and more effective buying and selling tactics, benefiting the broader economy and individuals within the private housing market.

3 Approach

Our strategy encompasses Exploratory Data Analysis (EDA), conventional time series forecasting, and cutting-edge Machine Learning (ML) models for nowcasting, leveraging an extensive dataset that reflects the property market's complexity. We meticulously analyze Property Resale Indices, their Percentage Changes, Construction Material Costs, Economic and Financial Indicators (including Bonds), and Prices Per Square Foot (PSF) to provide a nuanced understanding and accurate forecasts. The purpose of the ShinyApp is to provide users with an easy-to-use and comprehensive platform for analyzing the Property Price Index (PPI) time series, including capabilities for both forecasting and nowcasting. It offers tools for users to explore and compare these two approaches effectively.

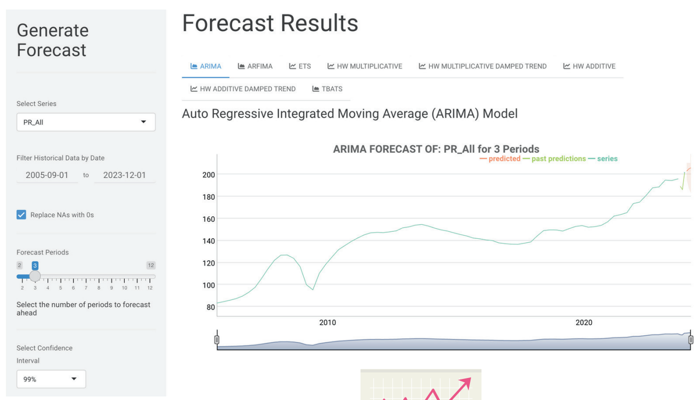
4 Exploratory Data Analysis



The Shiny App uses Exploratory Data Analysis (EDA) to give users a detailed view of the property market's current trends and relationships. With interactive visualisations for trends, distributions, and correlations, it equips users to better understand market dynamics.

- **Trend Analysis:**
  - Options to view line trends or boxplot trends.
  - Filters for time frequency (monthly or quarterly data).
  - Category and variable selection.
  - Data type options (original values or changes).
  - Date range selection (from start year and quarter to end year and quarter).
- **Distribution Analysis:**
  - Histograms to visualise variable distributions.
  - Filterable by category and variable.
  - Adjustable date range for targeted analysis.
- **Autocorrelation Analysis:**
  - Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) plots.
  - Category and variable selection, with data transformation options (none, log, square root).
  - Customizable lag settings for detailed lag analysis.
  - Confidence level adjustment for statistical significance.
  - Differencing order to identify data stationarity.
- **Crosscorrelation Analysis:**
  - Correlation Matrix to assess relationships between multiple variables.
  - Variable selection for comparative analysis.
  - Correlation type choices (Pearson, Spearman, Kendall).
  - Data transformation options to normalize or adjust variable scales.

5 Conventional Forecasting

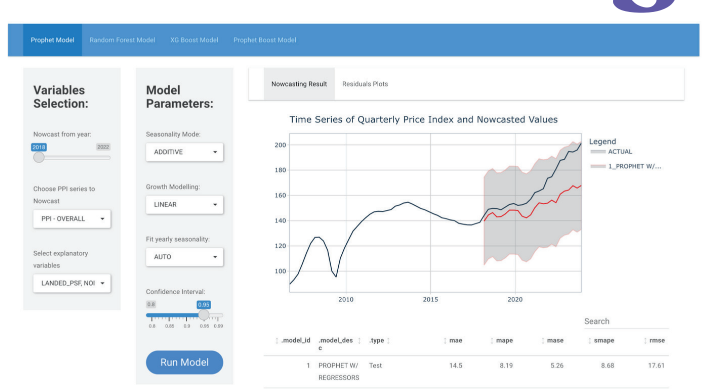


The Shiny App integrates conventional time series forecasting to provide users with reliable predictions of future market trends grounded in historical data. These time-tested models offer a critical perspective on potential market directions, essential for informed decision-making and strategy development in real estate. Coupled with EDA and nowcasting within the app, users gain a full spectrum of analytical tools for a robust market understanding.

- **Conventional Forecasting Models**:
  - ARIMA
  - ARFIMA
  - ETS (Exponential Smoothing State Space Model).
  - HW Multiplicative
  - HW Multiplicative Damped Trend.
  - HW Additive
  - HW Additive Damped Trend.
  - TBATS (Trigonometric, Box-Cox transformation, ARMA errors, Trend and Seasonal components).
- **User Customization**:
  - Variable filtering for targeted forecasting.
  - Start and end date selection for time range specificity.
  - Option to replace missing values (NAs) with zeros for continuity.
  - Forecast period adjustment (ranging from 2 to 12 periods).
  - Confidence interval settings for forecast precision.
  - Holdout period configuration (ranging from 2 to 12 periods) for model validation.

6 ML Models Nowcasting

MACHINE LEARNING



Models Overview

- **Tree-Based Models:**
  a. Random Forest Model: Averages multiple decision trees to reduce overfitting and improve predictive accuracy.
  b. XGBoost Model: Employs a more refined gradient boosting algorithm for efficient and effective performance.
- **Prophet-Based Models:**
  c. Prophet Model: Decomposes time series data into trend, seasonality, and holiday components, adaptable to different seasonality patterns.
  d. Prophet Boost Model: Integrates the decompositional approach of Prophet with the machine learning prowess of XGBoost.

Model Parameters Selection:

- **For Tree-Based Models:**
  - Random Forest: Parameters to customize include the number of trees for robustness, the number of variables considered for best splits to control diversity, and node size to determine the depth of trees.
  - XGBoost: Offers a comprehensive set of tunable parameters such as the number of boosted trees for sequential improvement, learning rate for convergence, and depth of each tree to prevent overfitting.
- **For Prophet-Based Models:**
  - Prophet: Adjusts parameters like seasonality mode (additive or multiplicative), growth model choice (linear or logistic), and yearly seasonality to capture inherent patterns.
  - Prophet Boost: Inherits Prophet's parameters while also leveraging XGBoost's learning rate and tree complexity to refine its predictive output.

Shared Features Across Models:

- Both categories allow for the selection of the starting year and PPI series variables, providing the option to include explanatory variables relevant to the nowcasting.
- Outputs from each model provide a combination of nowcasting results, with tree-based models additionally offering insights into variable importance, and all models including residual analysis for performance assessment.

7 Future Work

Future work may explore adding neural network and ensemble models, and the option for users to upload their own time series data for personalized analysis.