Introduction to Data Science 112

2nd Semester

Final Project Report

**Taiwan Real Estate Price Prediction Model**

**(Web Application)**

Author: Zenroy Chance

Student No.: 410921338

Due Date: 15.06.23

**System Name:** Taiwan Real Estate Price Prediction Application

**Aim:** To develop an accurate and reliable machine learning model that predicts real estate prices in Taiwan. My main goal is to assist buyers, sellers, and investors in making informed decisions by providing accurate price estimates for residential and commercial properties across different regions in Taiwan.

**Value:** This application is important because real estate buyers and investors can save significant time and effort by leveraging a price prediction model. Instead of manually researching and analysing numerous properties, they can use the model to quickly filter and prioritize options based on predicted prices. This efficiency can lead to cost savings in terms reduced time, effort and resources. In addition, this application can enable buyers, sellers, and investors to make informed decisions when it comes to purchasing or selling properties. By having reliable estimates of property prices, individuals can negotiate better deals and avoid overpaying or underselling.

**Relative Systems & Functions:** Relative Systems based their models on places like the US. These system are often developed to address similar objectives of predicting and analysing real estate prices. However, this has its unique characteristics and focuses specifically on predicting real estate prices in Taipei, Taiwan. Magnifying this small international country with its unique data set built into a model with attention to detail at the forefront makes this application different from it’s competition. In hindsight you can’t really compare Taiwan real estate price prediction to any other application just based on the uniqueness of the dataset.

**System Description:**

The first and hardest part of this project was finding data set of Taiwan Housing market which includes prices, location, number of rooms, size of lot etc. However I eventually came across this data set from <https://www.mohw.gov.tw/lp-130-2.html> which got this moving. After retrieving the dataset, the first thing on the list was to clean the data. Using numpy and matplotlib I used simple logic programming to remove outliers and drop columns from the dataset. To remove some the outliers, I used tables to see values that were on the far spectrum of the dataset. I also removed rare occurrences like 1 one-bedroom apartment being more expensive than a 3-bedroom apartment. After removing the outliers, I then built the model using sklearn and linear regression. Which brings me to the second largest challenge I faced; GridSearchCV for hyper parameters. My predictions were constantly producing negative results and sometimes. I would would get a warning on from my jupyter notebook.  
A screenshot of a computer

Description automatically generated

I came across an article on stackoverflow suggesting that I should set X and y to X.values and y.values and this was because the y already had values however X had column titles attach to it so I need to remove the columns so my train and test values would be equal.  *Reference:* [*https://stackoverflow.com/questions/69326639/sklearn-warning-valid-feature-names-in-version-1-0*](https://stackoverflow.com/questions/69326639/sklearn-warning-valid-feature-names-in-version-1-0)

A screenshot of a computer program

Description automatically generated with low confidence  
Finally, I then wrote a python flask server that uses the saved model to serve http requests. I then built a basic website built in HTML, CSS and JavaScript that allows user to enter home square ft area, bedrooms etc and it will call python flask server to retrieve the predicted price.

**Results:**

**A screenshot of a phone

Description automatically generated with medium confidence A screenshot of a phone

Description automatically generated with medium confidence**

**A screenshot of a phone

Description automatically generated with medium confidence** **A screenshot of a phone

Description automatically generated**

**A screenshot of a computer

Description automatically generated with medium confidence**

**A picture containing screenshot, rectangle

Description automatically generated**

**A picture containing screenshot, text, rectangle, diagram

Description automatically generated**

**Breakdown**

* Graphs were used as visualization tools to present statistical data. In this project they were used to determine what values were outliers.
* Data Processing and Model Building
  + - Python: The programming language used for the project.
    - Numpy and Pandas: Libraries for data cleaning, manipulation, and preprocessing.
    - Matplotlib: Library for data visualization and exploration.
    - Scikit-learn (sklearn): A popular machine learning library used for model building.
    - Linear Regression
    - Lasso Regression
    - Decision Tree
* Flask Framework was used as a medium from transforming the prediction model into a web application. Flask provided the essentials needed to create my web app by allowing me to used functions such as routing URLs, handling HTTP requests and responses.
* HTML, CSS and JavaScript was used to build web application in which clients/users can enter parameters into the model and providing them with instant feedback of predicted prices. HTML was used for Structure of the webpage, CSS for the design and styling of the page and JS (AJAX) sent and received Json responses between the server and the client.
* For live testing and Debugging Postman was used to test post and get requestion on a live environment  
  A screenshot of a computer

  Description automatically generated

A screen shot of a computer program

Description automatically generated with medium confidence

**Conclusion:**

The project focused on predicting housing prices in Taiwan using a dataset obtained from the Ministry of Health and Welfare website. The data cleaning process involved removing outliers and dropping irrelevant columns. The prediction model was built using linear regression from the sklearn library, with the challenge of tuning hyperparameters using GridSearchCV. A Python Flask server was developed to serve HTTP requests and retrieve predicted prices, and a basic website was created using HTML, CSS, and JavaScript to provide a user interface for entering property details. Despite challenges, the project successfully combined data science techniques with web development to enable housing price predictions in Taiwan.