**Introduction**

With the increasing demand for purchasing real estate, people are looking forward to knowing the approximate price of their dream properties. The purpose of the project is to design property management and price prediction software based on Java. It incorporates SQLite database storage, Jlist display, and a predictor based on the KNN algorithm. The project takes a list of NYC property sale prices as an example.

**Design**

1. Property Database Manager

It creates a property table, consisting of type, price, the number of beds and baths, size, and the location of the property. It reads from NYC housing price CSV and outputs propertydata.db as shown below.

图形用户界面, 文本

描述已自动生成

Figure 1

The manager can also prevent CSV lines with the wrong format. 图形用户界面, 应用程序, 表格

描述已自动生成

Figure 2

1. Property display

The Jlist allows users to filter the results by the least number of baths, beds, square feet, housing types, and locations. JComBox and JTextField are used to minimize filtering steps and improve UI experience. Also, an extra button is displayed for KNN prediction initiation.

表格

描述已自动生成图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成表格

描述已自动生成

Figure 3

1. Price Predictor

The number of baths, beds, square feet, housing types, and locations are used as KNN training parameters. In KNN, the Euclidean distance of the input and the database data is calculated. After ranking the distance, the least K number of data's average price is used as the final prediction. To avoid overfitting and underfitting, K = 5 is used.

Euclidean distance is calculated by,

Meanwhile, the borough location and housing types are added to the algorithm through penalty terms. It happens when the property types or boroughs do not match the input. For example, if other parameters are the same and the input type is house and the database’s condo, an additional 10000 distance penalty is added. However, if the types are the same, no penalty is needed.

图形用户界面, 应用程序

描述已自动生成 图形用户界面, 表格

描述已自动生成

Figure 4

**Discussion**

There are several potential optimizations. First, inserting CSV lines in batches can be faster when it comes to a large database. Second, JavaFX can be used as a UI tool to ensure a smooth and modern user experience. Also, property statistical graphs such as Brooklyn Property Price Distribution based on Types can be visualized for data display.

**Conclusion**

The project successfully conducts data management, filtering, prediction, and display to assist potential asset buyers. SQLite and Java GUI are utilized to ensure efficient user experience. The database can also vary to meet the needs of other cities.