**Comparative Analysis of Regression Algorithms for House Price per Square Foot**

**Introduction:**

The real estate market is dynamic and influenced by various factors that contribute to the fluctuation of house prices. In this project, we aim to predict the price per square foot of houses using machine learning algorithms. Accurate predictions can assist potential buyers, sellers, and real estate professionals in making informed decisions.

**Objective:**

The objective of the project is to develop a predictive model that can predict the price per square foot of a house based on the area, location and its features.

**Methodology:**

* **Data Collection:** The data used to build the model was provided in the csv format. It has different columns such as price, price per square foot, location, availability, area type, number of bedrooms and bathrooms, etc.
* **Data Pre-processing:** Data pre-processing is a crucial step to ensure the quality and suitability of the dataset for training machine learning models.
* **Feature Selection:** Feature selection is a critical step to identify the most relevant variables that contribute to the predictive power of the model.
* **Model Selection:** In the model selection section, provide a detailed overview of the machine learning algorithms chosen for the predictive analysis. Explain the rationale behind the selection of each algorithm and discuss how they align with the project objectives.
* **Model Training:** In the model training section, the processed data is fit to train the selected model so that it is able to predict the future entered data.
* **Model Evaluation:** In the model evaluation section, the performance of the trained machine learning models is assessed to select the best suited model for deployment.

**Algorithm Used:**

* **KNeighborsRegressor:** KNeighborsRegressor is a machine learning algorithm used for regression tasks. It is part of the scikit-learn library in Python and is based on the K-Nearest Neighbors (KNN) approach, similar to the KNeighborsClassifier for classification tasks. The KNN algorithm works by predicting the target variable of a new data point based on the average (or weighted average) of the target values of its k-nearest neighbors in the feature space.
* **Linear Regression:** Linear Regression is a statistical method and a fundamental machine learning algorithm used for predicting a continuous outcome variable (also called the dependent variable) based on one or more predictor variables (independent variables). The relationship between the variables is assumed to be linear, meaning that a change in the predictor variables is associated with a linear change in the outcome.
* **Ridge:** Ridge Regression, also known as Tikhonov regularization or L2 regularization, is a linear regression variant that introduces a regularization term to the standard linear regression objective function. The purpose of Ridge Regression is to prevent overfitting and handle multicollinearity, which occurs when predictor variables are highly correlated.
* **Lasso:** Lasso Regression, short for Least Absolute Shrinkage and Selection Operator, is a linear regression technique that introduces a regularization term to the standard linear regression objective function. Like Ridge Regression, Lasso aims to prevent overfitting and handle multicollinearity in the presence of highly correlated predictor variables.
* **Polynomial Regression:** Polynomial Regression is a type of linear regression in which the relationship between the independent variable (predictor variable) and the dependent variable is modeled as an nth-degree polynomial. It extends the simple linear regression model, allowing for a more flexible representation of the relationship between variables.

**Solution Architecture:**

* **Data Collection and Storage:** The data for the project was provided to us in a csv format. Using ‘pd.read\_csv()’ the data was used to read in the jupyter notebook for further processing.
* **Data Pre-processing:** Working with dataset like this with 7120 rows, the pre-processing stage, is important to look for the missing or null values as well as duplicated data which might make the model less accurate. In this case the dataset has no null values.
* **Feature Selection:** The Bangalore house price data column were already encoded and was easy to select the features for model training such as location, availability, area type, area , number of bedrooms and bathrooms.
* **Model Selection:** Model selection is based on the type of dataset or taking the problem in mind. Here we need to predict the price per square feet of the house, so, it is a regression problem and requires regression models such as KNeighborsRegressor, Linear Regression, Ridge, Lasso and Polynomial Regression.
* **Model Training:** Then the models are fitted on the dataset and trained to predict for the test dataset.
* **Model Evaluation:** Next step requires for the evaluation of the model for its accuracy using different metrics such as r2\_score, r2\_accuracy, precision\_score, etc. Here, on evaluation the KNeighborsRegressor gave an accuracy of 98% , Linear Regression model gave an accuracy of 69%, Ridge model gave an accuracy of 68%, Lasso model gave an accuracy of 68% and Random Forest model gave an accuracy of o 81%.
* **Best Model Selection:** Then the model with highest accuracy is selected and saved for future use.