ECS171 Final Report

ZongYuan Wei, Peter Qu, Saiyudi Ma, Yitian Zhang, Yixin Mao

June 2024

1 Introduction & Background

The housing price has always been a problem for buyers, picking the preferable housing has always been a challenge not only to buyers but also to real estate agents. Studying the data that pertains to the houses found in a given California district and some summary stats about them based on the 1990 census data, will give a sense of how buyers' information and their needs are related, also gathering that information will give real estate agent a general understanding of what they expect their future customer needs, and provide them with more preferable housing lists. Even though this data may not help people with predicting current housing prices, it may still give people a general trend aspect when they look through the analysis.

The primary aim for this dataset is to predict median house values in California districts based on various features. From the information obtained in the dataset, our group can perform a data analysis based on features such as median income, housing age, population metrics. This dataset has gained substantial popularity in machine learning due to its rich feature set and real-world applicability making it an ideal candidate for predictive modeling and exploratory data analysis. Additionally, by analyzing the data in the dataset, we can identify trends and patterns in the real estate market, to understand the needs of different sizes of families. Generally, families with large family members will pick on houses with more bedrooms and bathrooms, those often come with large square feet of living area size. On the contrary, families with fewer family members will have less preference for the number of rooms.

This dataset was created using data from 1990 census, the time when California was undergoing significant demographic and economic changes, many people lose their jobs which means their income will be less than other years, it will reflect on the median income data in the dataset. Even though the recession did not last that long but still impact the income in a way that is inevitable. This unique factor makes the dataset less accurate while studying current housing prices, but it provide a relevant comparison between different households.

Location of the house plays a vital role in determine the housing prices. The latitude and longitude coordinates allows for detailed geographic analysis. Visualize housing price trends across different regions, identify high and low-price areas made choosing housing more easier. This is crucial for real estate investment and understanding the regional economic disparities.

Extensively used in academic settings to teach various machine learning concepts, this datasets requires rudimentary data cleaning and it is helpful for students to implementing machine learning algorithms. Working with this dataset involves dealing with challenges related to data quality such as missing values, outliers harder. Especially requires handling the analysis and pick the proper model for the datasets more carefully. Data cleaning and preparation is needed before using high predictive accuracy model, but some often lack interpret ability. Plotting the datasets into suitable plots first would help doing analysis more easier.

The California Housing Prices dataset is a comprehensive dataset that serves as a robust resource that play a significant role in data analysis and machine learning. Its complicated set of features and real-world applicability make it ideal for predictive modeling, geographic analysis and educational purposes. Furthermore it helps real estate agents and buyers to understand the median prices trend in certain area during 1990s housing market. By providing a detailed analysis of California housing market in 1990, it offers a valuable insights into various factors that driving housing prices in California. Moreover, the practical challenges presented in data processing, model interpret ability makes it an excellent case study for analyze the sophisticated techniques to real-world data.

2 Literature Review

This California Housing Prices dataset is initially featured in the paper: Pace, R. Kelley, and Ronald Barry. "Sparse spatial auto regressions." Statistics and Probability Letters 33.3 (1997): 291-297. but later Aurélien Géron used same dataset in 'Hands-On Machine learning with Scikit-Learn and TensorFlow'. In Aurélien Géron's work, he wrote datasets is a modified version of California Housing dataset from Luís Torgo's page (University of Porto). The datasets is not cleaned, it has multiple 'Na' values. In order to use the datasets to perform analysis and perform the training on the sets to compare and figure out the proper model, we had to enhance the data by cleaning the invalid value first and then perform the data analysis. The datasets that we are analyzing is modified from Aurélien Géron's work, which differs by 207 values were randomly removed from Pace, R. Kelley, and Ronald Barry and add an additional categorical attribute ocean-proximity which enables a deeper analysis of categorical data and its impact on housing prices.

Since original datasets does not include any conclusion about how various attributes contributed to the median housing prices, our analysis extends Géron's work by applying various regression models to the cleaned dataset. we used visualizations and plots to figure out the relationships between features and median house value. By evaluates Mean Squared Error and coefficient of determination of different regression models, we assess the effectiveness of different predictive approaches.

We conducted experiments with multiple regression models including linear regression, polynomial regression, polynomial ridge regression and linear ridge regression. Compared each model's performance to determine the most effective approach to predicting housing prices. This helps to identify the strengths and weakness of each algorithms we used when applied to real-world datasets. The study of new categorical attribute also let our group examines the geographic distribution of housing prices. The coordinates of latitude and longitude enables a detailed maps visualizing trends and identifying regional disparities. This analysis provides an understanding of how location influences housing prices. By incorporating new features and cleaning the data, our group gain deeper insights to the factors affecting housing prices.

3 Dataset Description and Exploratory Data Analysis

The dataset contains the houses in a given California district and some basic info on the 1990 census data. There are 10 columns: **Longitude&Latitude**: the measure of how far west/north the house is. **HousingMedianAge**: median age of the house within a block, a lower number indicates a newer building. **total_rooms**: total number of rooms within a block. **total_bedrooms**: total number of bedrooms within a block. **population**: total number of people residing within a block. **households**: total number of households, a group of people residing within a home unit. **median_income**: median income for households within a block. **median_house_value**: median house value for households within a block. **ocean_proximity**: with ocean/sea nearby. Here are all the descriptions of each variable feature, cited from `https://www.kaggle.com/datasets/camnugent/california-housing-prices`. We will use the given data to train our model with these variables and predict the median house value.

3.1 Invalid value cleaning

First, the data-cleaning procedure takes action; we check whether 'Na' values exist in the data and find that in the column "total_bedrooms", there are 207 'Na' values. Our first intuition is that the total number of bedrooms is zero in these situations, but after checking the whole dataset, they are just invalid data points. After, since 207 out of 20640 rows is a reasonable number to delete, we remove all those values in case making sure there are enough data points left.

3.2 longitude and latitude

Next, we look at the position of the houses and check the distribution of the longitude and latitude. Most latitude values are around 34.2, with an upper limit of 42 and a lower limit of 33.5. Likewise, longitude ranges from -114.2 to -124.2. Most focus on -118.5. Because this number only represents location information, its magnitude does not represent any real meaning. So we plot the position of the housing with respect to it's value. (figure 1)

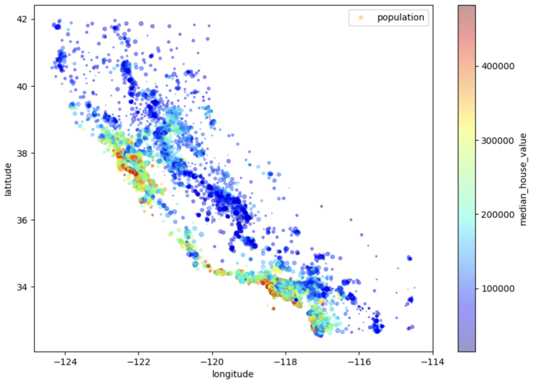


Figure 1: geography info

3.3 Distribution plot of the variables

Then we also simply plot the distribution of each numerical variables to give a sense of them. First, we look at the distribution of the total number of rooms, most of which are concentrated in 0-500, ranging from 1 to 6445), and then the number of people in the block is concentrated within 3000, with a minimum value of 3 and a maximum of 35,682. The average is 1424. household numbers are also concentrated within 600, ranging from 1 to 600. Average income is a little more widely distributed, being more evenly distributed across income ranges. Finally, our predicted house prices are distributed more widely, ranging from 14,999 to 206,864. But from the distribution map, there are actually a lot of extreme numbers in the data, they affect the range of data, and they are far from the average, and the number is very small. So the problem of deleting them will be mentioned later.

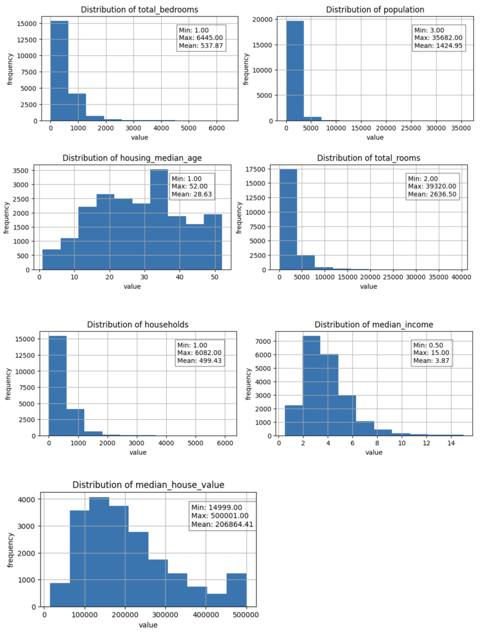


Figure 2: distribution of variables and correlation plot

3.4 Check whether two or more of the existing variables are related:

The correlation plot above shows well that totalbedroom has the highest correlation with totalrooms, which is also very easy to explain, followed by the correlation between totalrooms and household. Then there's the correlation between population and totalrooms, and then there's the correlation between household and population. The remaining correlations are less significant. However, if we want to directly delete these variables with very high correlation, we will encounter such problems: 1: our existing data is not enough, whether directly deleting the whole variable will cause information loss. So keep these variables first, and when we train the model, we can try to drop different variables to see if the model is overfit.

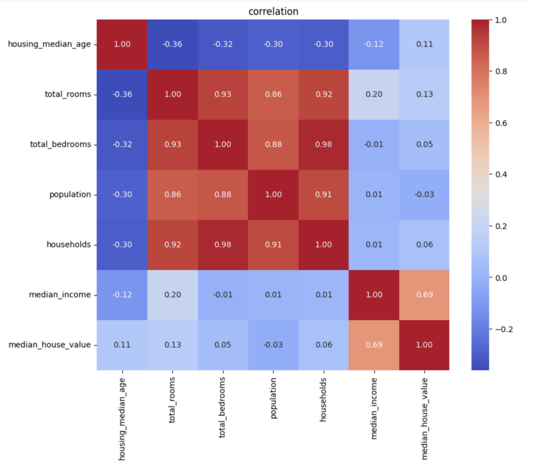


Figure 3: correlation plot

3.5 Outliers:

Finally, we checked the situation of outlier, because we could clearly see the existence of extreme values from the previous distribution diagram, so I found 2999 Outliers by statistical method, those points exceed 1.5 of IQR are seens as outliers, then deleted them, and noe the size of the cleared dataset was 17434x10, which is cleaned to handle and didn't way less than the original dataset.

4 Proposed methodology

4.1 Data Preprocessing

4.1.1 Data Cleaning

We began by removing rows with missing values in the `total_bedrooms` column, reducing the dataset size from 20,640 to 20,433 rows. Outliers were detected and removed using the Interquartile Range (IQR) method, which further reduced the dataset to 17,434 rows.

4.1.2 Feature Engineering

Categorical variables (`ocean_proximity`) were encoded using one-hot encoding to convert them into a numerical format suitable for machine learning algorithms. The dataset was then split into training and testing sets with an 80:20 ratio to evaluate model performance on unseen data.

4.1.3 Feature Scaling

StandardScaler was applied to normalize the feature set, ensuring that the model was not biased towards variables with higher magnitudes.

4.2 Model Selection

4.2.1 Linear Regression

Linear Regression served as the baseline model due to its simplicity and ease of interpretation. However, it struggled to capture non-linear relationships in the data.

4.2.2 Polynomial Regression

We increased the complexity of the model by adding polynomial features. Initially, this model suffered from severe overfitting, indicated by an extremely high Mean Squared Error (MSE) and negative R^2 on the test data.

4.2.3 Ridge Regression

To reduce overfitting and improve model generalizability, we introduced Ridge Regression, which penalizes large coefficients. Two variants were tested: Linear Ridge Regression and Polynomial Ridge Regression.

4.2.4 Polynomial Ridge Regression

Combining polynomial features with Ridge regularization allowed us to capture non-linear patterns while controlling overfitting. This model demonstrated superior performance compared to others, with the lowest MSE and highest R^2.

4.3 Model Evaluation

4.3.1 Metrics

Mean Squared Error (MSE) and the coefficient of determination (R^2) were used as primary evaluation metrics to assess predictive accuracy and explanatory power.

4.3.2 Cross-Validation

We employed K-Fold cross-validation to ensure model robustness and generalizability, evaluating average MSE and R^2 scores across multiple folds.

4.3.3 Learning Curves

Learning curves were analyzed to understand the model's performance as the training size increased. Polynomial Ridge Regression showed the best balance between bias and variance, indicating its effectiveness in capturing data complexity.

4.4 Implementation Steps

4.4.1 Import Libraries and Dataset

We began by importing the necessary libraries for data manipulation, visualization, and model training. The dataset was then loaded from a CSV file.

4.4.2 Load and Clean Data

The dataset was inspected for missing values and outliers. Missing values in the `total_bedrooms` column were removed, and outliers were detected and removed using the Interquartile Range (IQR) method.

4.4.3 Feature Engineering and Scaling

Categorical variables, such as `ocean_proximity`, were encoded using one-hot encoding. The dataset was split into training and testing sets in an 80:20 ratio. Feature scaling was applied using StandardScaler to normalize the feature set.

4.4.4 Model Training and Evaluation

Several models were trained and evaluated. Linear Regression was used as a baseline model. Polynomial features were added, and Ridge Regression was applied to reduce overfitting. Models were evaluated using Mean Squared Error (MSE) and the coefficient of determination (R^2).

4.4.5 Visualizations

Various visualizations were created to analyze the data and model performance. Scatter plots were used to compare actual vs. predicted values for different models. Learning curves were generated to understand the model's performance as the training size increased.

5 Experimental results and evaluation

5.1 Experimental Setup:

To evaluate the performance of various regression models, we conducted a series of experiments using a cleaned dataset. This dataset was prepared by encoding categorical variables, splitting it into training and testing sets, and scaling the features. The primary focus was on comparing the following models:

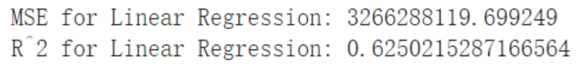
1. **Linear Regression**
2. **Polynomial Regression**
3. **Polynomial Ridge Regression**
4. **Linear Ridge Regression**

5.2 Metrics used:

The models were assessed using Mean Squared Error (MSE) and the coefficient of determination (R^2) as the primary metrics for evaluation. These metrics provide insight into the predictive accuracy and explanatory power of the models:

5.3 Linear Regression Performance:

Linear Regression was used as a baseline model due to its simplicity and ease of interpretation. The results were:



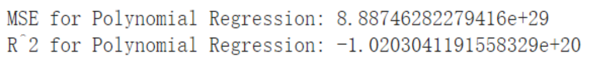
```
MSE for Linear Regression: 3266288119.699249
R^2 for Linear Regression: 0.6250215287166564
```

Figure 4: Linear Regression: MSE vs R^2

These results indicate that while Linear Regression can explain a substantial portion of the variance in the dataset (0.625), there is still a significant amount of error in its predictions, as reflected by the high MSE.

5.4 Polynomial Regression Performance:

To explore non-linear relationships in the data, we introduced polynomial features. The Polynomial Regression model, however, may suffered from overfitting, as evidenced by its performance metrics:



```
MSE for Polynomial Regression: 8.88746282279416e+29
R^2 for Polynomial Regression: -1.0203041191558329e+20
```

Figure 5: Polynomial Regression: MSE vs R^2

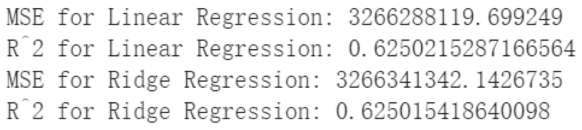
The extreme high MSE and negative R^2 suggest that the model fits the training data excessively well but fails to generalize to the testing data. This overfitting is likely due to the high degree of the polynomial terms introduced without any regularization.

5.5 Explanation of Ridge Regression Utilization:

The shortcomings observed in both Linear and Polynomial Regression models necessitated the exploration of regularization techniques. While Lasso Regression could potentially address multicollinearity and overfitting by imposing sparsity on the coefficient estimates, it may excessively penalize certain features, leading to feature selection and potentially oversimplified models. Ridge Regression, on the other hand, offers a balanced approach by penalizing large coefficients without completely eliminating any predictors. By introducing a regularization term to the loss function, Ridge Regression effectively mitigates multicollinearity and reduces overfitting, thus enhancing the model's generalizability and predictive performance. Consequently, the decision to employ Ridge Regression stemmed from its ability to strike a balance between model complexity and predictive accuracy, making it a suitable choice for improving the stability and interpretability of the model.

5.6 Linear Ridge Regression Performance:

we trained a Linear Regression model with Ridge regularization to examine if regularization alone, without polynomial features, could improve performance:



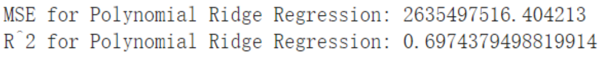
```
MSE for Linear Regression: 3266288119.699249
R^2 for Linear Regression: 0.6250215287166564
MSE for Ridge Regression: 3266341342.1426735
R^2 for Ridge Regression: 0.625015418640098
```

Figure 6: Linear Regression vs Linear Ridge Regression

The Linear Ridge Regression model performed similarly to the standard Linear Regression model, suggesting that regularization had a minimal impact in this scenario. This indicates that the model's performance was more limited by its linear nature rather than by overfitting.

5.7 Polynomial Ridge Regression Performance:

To mitigate the overfitting observed in Polynomial Regression, we applied Ridge regularization, which penalizes large coefficients in the polynomial terms. This approach improved the model's performance:



```
MSE for Polynomial Ridge Regression: 2635497516.404213
R^2 for Polynomial Ridge Regression: 0.6974379498819914
```

Figure 7: Polynomial Ridge Regression: MSE vs R^2

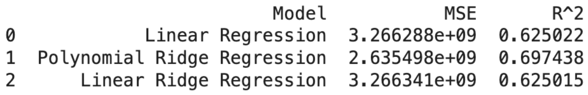
The Polynomial Ridge Regression model demonstrated a substantial reduction in MSE and an increase in R^2 compared to the Linear Regression model, indicating a better fit and predictive accuracy.

5.8 Evaluation and Comparison:

Compare MSE and R2

Because the Polynomial Regression model performed particularly poorly, with extremely high MSE and negative R^2 values, we decided to exclude it from further comparison.

Based on the evaluation results of those three models we can see that polynomial Ridge Regression has the lowest MSE, which means that indicating the best predictive accuracy. And both Linear Regression and Linear Ridge Regression have a pretty similar MSE and are larger than the polynomial ridge, suggesting that they struggle to capture data complexity.



```
                          Model           MSE       R^2
0             Linear Regression  3.266288e+09  0.625022
1  Polynomial Ridge Regression  2.635498e+09  0.697438
2       Linear Ridge Regression  3.266341e+09  0.625015
```

Figure 8: All Three Models MSE vs R^2

For the R^2, Polynomial Ridge Regression also demonstrates the highest R^2 value at 0.697, implying the best explanatory power for data variance. Linear and Linear Ridge have R^2 values of 0.625, reflecting moderate performance.

Linear Regression is a simpler model with high MSE and moderate R^2, indicating its limited capacity to capture non-linear relationships. Linear Ridge Regression performs similarly to Linear Regression, indicating that regularization does not significantly impact this model.

Polynomial Ridge Regression shows better MSE and R^2, suggesting it better captures non-linear patterns in the data, but it might require more computational resources.

Gradient Descent

The evaluation results after applying Gradient Descent optimization indicate that the Polynomial Ridge Regression model outperforms the other models. Linear Regression and Linear Ridge Regression show very similar performance, suggesting that regularization does not significantly improve the linear model's performance.

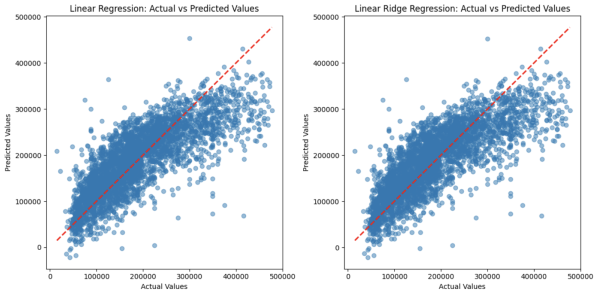


Figure 9: Linear and Linear Ridge

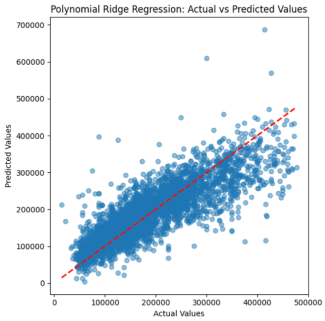


Figure 10: Polynomial Ridge

In contrast, the Polynomial Ridge Regression model, optimized using GD, effectively captures non-linear relationships, resulting in superior predictive accuracy and explanatory power. The scatter plots confirm that the predicted values from Polynomial Ridge Regression are closer to the actual values compared to the other models.

Therefore, after GD optimization, the Polynomial Ridge Regression model stands out as the most suitable choice, providing enhanced performance and effectively capturing the complexity and non-linear patterns in the data.

K-Folds Cross

Based on the evaluation results using KFold cross-validation, the Polynomial Ridge Regression model performs the best among the three models tested. The Linear Regression and Linear Ridge Regression models have very similar performance, indicating that adding regularization does not significantly impact the linear model's effectiveness.

KFold cross-validation results show that the Polynomial Ridge Regression model captures non-linear relationships more effectively, leading to improved predictive accuracy and explanatory power. The scatter plots confirm this, with predicted values from Polynomial Ridge Regression closer to actual values compared to the other models.

Learning Curve

Based on the learning curves, we can draw several key conclusions about the performance of the three models.

The learning curves for Linear Regression and Linear Ridge Regression are quite similar. Both models exhibit high training and cross-validation errors, indicating underfitting. The training errors start around 3.35e9 and slightly decrease, stabilizing around 3.3e9 as the number of training examples increases.
The cross-validation errors also start around 3.35e9 and remain relatively flat, indicating that adding more data would not significantly improve their performance. Regularization in the Linear Ridge model does not have a significant impact, as the learning curves for both models are almost identical, suggesting that both models struggle to capture the underlying complexity of the data.

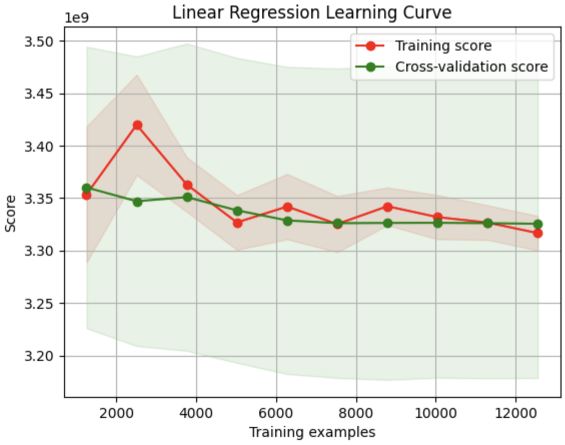


Figure 11: Linear Regression Learning Curve

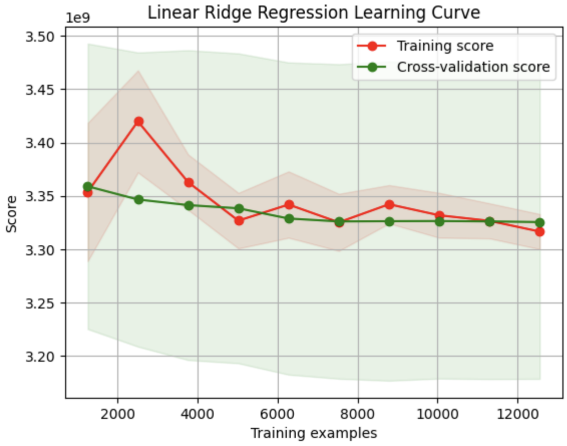


Figure 12: Linear Ridge Learning Curve

In contrast, the Polynomial Ridge Regression model shows a more favorable learning curve. The training error starts at approximately 2.7e9 and decreases as more training examples

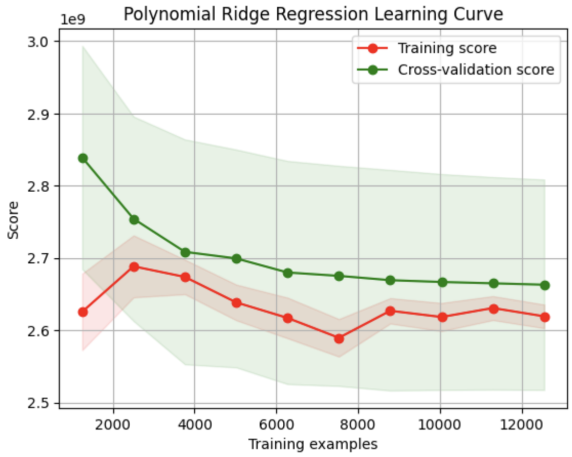


Figure 13: Polynomial Ridge Learning Curve

are added, stabilizing around 2.6e9. The cross-validation error starts higher at around 2.9e9 but gradually decreases and converges towards the training error, stabilizing around 2.7e9. This convergence indicates improved generalization and suggests that the model benefits from additional training data. The smaller gap between the training and cross-validation errors indicates a better balance between bias and variance.

In conclusion, the Polynomial Ridge Regression model outperforms the other two models by effectively capturing non-linear patterns in the data. It benefits more from additional data, leading to reduced training and validation errors. The learning curves highlight its superior performance and better generalization capabilities, making it the most suitable model for this dataset.

6 conclusion and discussion

To conclude, our analysis of the Housing Price Dataset has provided valuable insights into real estate trends and patterns. By leveraging machine learning models and techniques, we were able to predict house prices with reasonable accuracy and understand the underlying factors influencing housing values. Through exploratory data analysis, we uncovered relationships between various features such as location, amenities, and demographic factors, shedding light on the preferences and behaviors of homebuyers.

Our project highlighted the importance of teamwork and collaboration in achieving project goals. Effective communication and active participation among team members were essential to our success. By dividing tasks and leveraging each member's unique strengths, we simplified the workflow and delivered high-quality results within the given time period. Team collaboration was critical, as each member brought diverse skills and perspectives to the table. Coordinated efforts ensured we met our milestones and produced a cohesive final report. This project not only deepened our understanding of data analysis and machine learning but also highlighted the value of teamwork in achieving complex objectives.

Moving forward, there are several areas for further exploration and improvement. We can refine our feature selection process to enhance model interpretability and uncover more meaningful insights from the data. Additionally, incorporating advanced techniques such as ensemble learning or deep neural networks may offer new insights and opportunities for modeling complex relationships. These approaches could help us gain a deeper understanding of the housing market trends and improve our ability to predict future market behaviors.

Overall, we are all proud of the work we have accomplished and excited to share our findings. By sharing our research findings and contributing to the body of knowledge in the field of real estate analytics, we hope to inform future research and decision-making processes in the housing market.

7 Github link and Project roadmap and any comments about assignment of tasks to team members

Github link:

`https://github.com/VincentWei0120/ECS171-Project`

Project roadmap:

Week 4-5: Problem Understanding

- Review and understand the current features in housing price prediction.

- Explore and review the dataset to identify relevant features and trends.

Week 6-7: Model Training

- Preprocess/clean the data and handle extreme values.
- Train data and develop predictive models using methods learned in class (e.g., linear regression, polynomial regression).
- Evaluate the model performance using methods learned in class (e.g., Mean Squared Error (MSE), Gradient Descent, k-fold cross-validation).

Week 8: Analysis & Visualization

- Analyze and visualize the data trends and patterns.
- Identify the factors that relate to our problem statement.

Week 9: Writing Report

- Write a report that follows the specific format and guidelines.

Week 10: Remaining Week

- Finish previous parts that haven't been completed yet.
- Double-check and submit.

Comments:
Well done guys! Happy Summer.

References

[1] "1998 Cal Facts: California's Economy," [Online]. Available: https://lao.ca.gov/1998/1998_calfacts/98calfacts_economy.html. [Accessed: Jun. 8, 2024].

[2] C. Nugent, "California Housing Prices Dataset," Kaggle, 2017. [Online]. Available: https://www.kaggle.com/datasets/camnugent/california-housing-prices. [Accessed: Jun. 8, 2024].

[3] A. Géron, "Hands-On Machine Learning with Scikit-Learn and TensorFlow," [Online]. Available: https://github.com/ageron/handson-ml/tree/master/datasets/housing. [Accessed: Jun. 8, 2024].