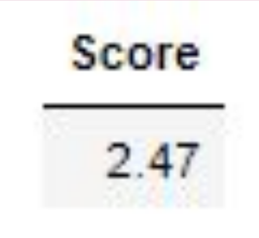
Kaggle Project: House Prices - Advanced Regression Techniques

Kenya Galvan

Overview

- 80 features including lot shape, utilities, nearby allies, and more general data about the individual houses.
- Preprocessing
  - Removed columns containing many null values
  - Made categorical values into numeric ones
- 3 regression models using scikit-learn.
- Place: 4139



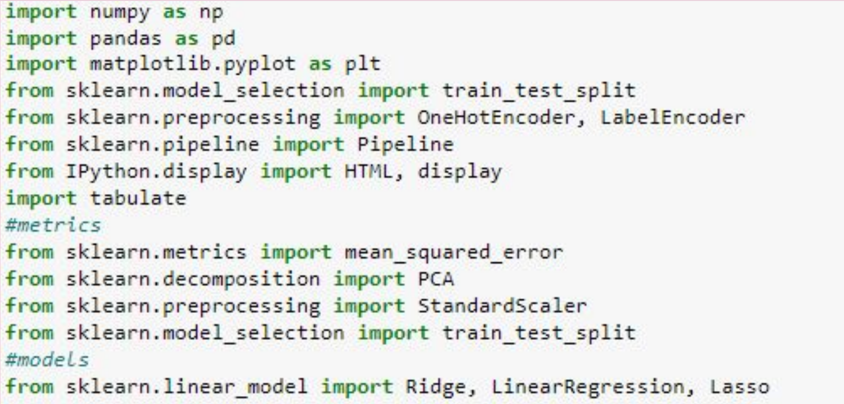
Score
2.47

Data

https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques

Predict sales prices given house information.

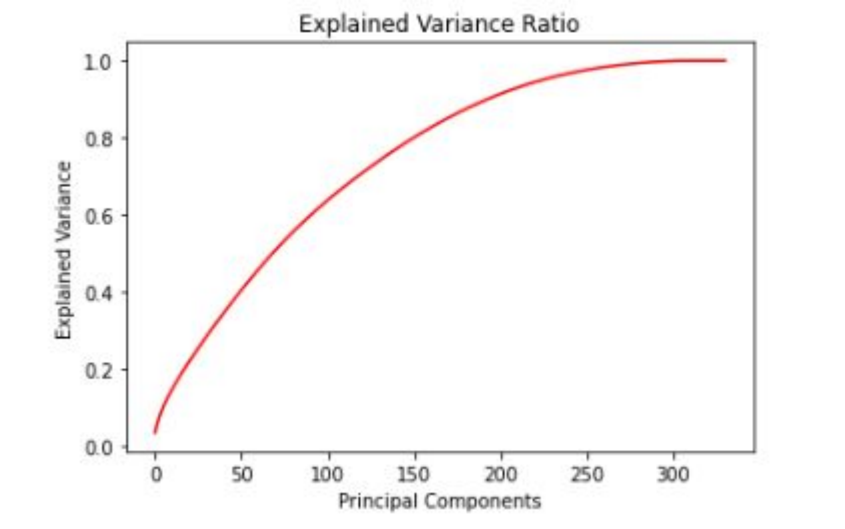
- Input: House information
  - CSV: 80 columns
  - Lot shape, land contours, utilities, street, nearby allies, lot areas, etc.
- Output: Sale price
  - 81th column
- Root-Mean_squared_error
- 1458 data points



```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import OneHotEncoder, LabelEncoder
from sklearn.pipeline import Pipeline
from IPython.display import HTML, display
import tabulate
#metrics
from sklearn.metrics import mean_squared_error
from sklearn.decomposition import PCA
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split
#models
from sklearn.linear_model import Ridge, LinearRegression, Lasso
```

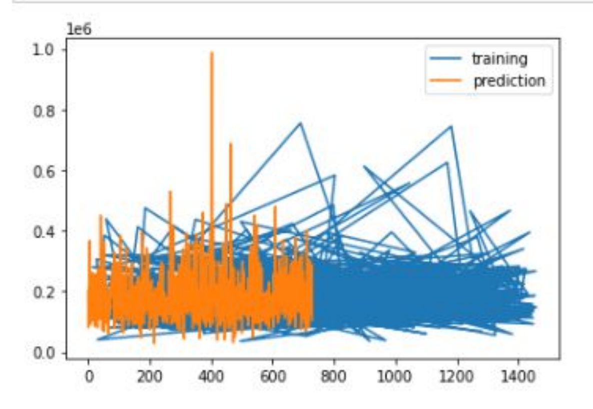
Visualization

- This tells us we could probably use 150-200 features and still have significant variance.



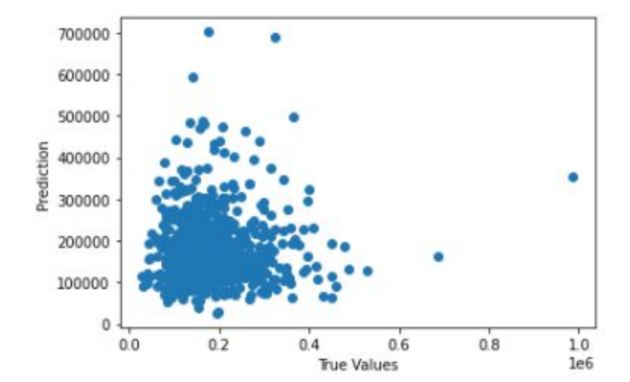
Ridge Regression:

- Predicted vs. Training
- How was the data captured?



Predictions:

- Bad linear regression.
- I should have made it linear.



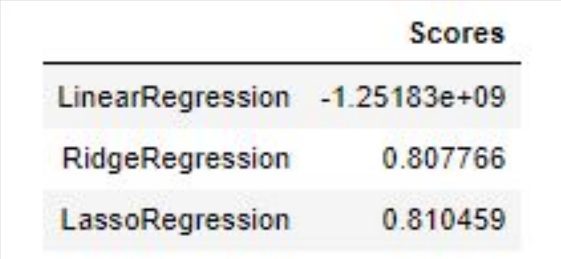
Training + Performance

- Scikit-learn

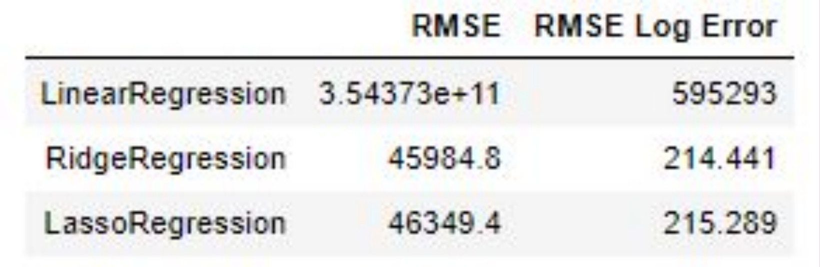
Lasso regression: To avoid overfitting

Ridge regression: I was most familiar with it

Linear regression: I was most familiar with it



	Scores
LinearRegression	-1.25183e+09
RidgeRegression	0.807766
LassoRegression	0.810459



	RMSE	RMSE Log Error
LinearRegression	3.54373e+11	595293
RidgeRegression	45984.8	214.441
LassoRegression	46349.4	215.289

Conclusion: **Linear** regression was the best model to use.