Assignment 3: Evaluating Regression Models by Mimi Trinh

Section 1: Summary and Problem Definition

The Boston housing study is a market response study including 506 census tracts in the Boston metropolitan area. The objective of the study is to advise a real estate brokerage firm in its attempt to employ machine learning methods to complement conventional methods for assessing the market value of residential real estate. The response variable is the median value of homes in thousands of 1970 dollars. The remaining variables in the dataset are predictors.

Section 2: Research Design, Measurement, and Statistical Methods

The study starts with 14 columns in the dataset, including the response variable and 13 predictors. However, the neighborhood column is dropped, so the dataset is narrowed down to 13 variables. There's no missing value, so the dataset is clean to be analyzed. First, standard scaler is implemented since it's best practice to standardize the variables before analysis. Second, an exploratory data analysis (EDA) is conducted to examine the response variable and correlation between the response variable and the predictors. Third, we build four models using linear regression, ridge regression, lasso regression, and elastic net. For each model, within a ten-fold cross validation design, we use the root mean squared error (RMSE) to evaluate the methods. In other words, the mean of 10 RMSE scores is an index of prediction error of each model.

Section 3: Programming Work

Multiple Python packages are utilized to do the programming: numpy, pandas, Scikit-Learn, and matplotlib. We start the project by feeding the csv raw data file into Python. Then we drop the neighborhood column from the dataset and start the standard scaler transformation process. Matplotlib is utilized to conduct the EDA to understand the data and correlation among variables. Next we use LinearRegression(), Ridge(), Lasso(), and ElasticNet() to build four models. For each model, we use cross_val_score() to design the ten-fold cross validation. Python doesn't have RMSE as a scoring metric by default, so we use neg_mean_squared_error as the scoring metric inside cross_val_score and np.sqrt() to convert it to RMSE. Finally, using intercept_ and coef_, we extract the intercept and coefficients to build an equation from the best model. Generally, we avoid linear regression. Ridge regression is a good default to start. In situation where we suspect only a few variables are significant, we typically start with Lasso regression or elastic net. In this case, we don't have industry knowledge of the dataset, so we build all four models and let the RMSE metric determines the best model.

Section 4: Results and Recommendations

Exhibit 1 shows that the dataset has no null record, so we don't have to address missing value issue. Exhibit 2 gives the descriptive statistics of the dataset as part of the EDA result. Exhibit 3 shows that the response variable has outliers, but there's no extreme outlier since there's no observation outside the +/- 3 standard deviation range. The variable is skewed positive, but since there's no extreme outlier, we recommend not to remove any outlier and continue with the study. Exhibit 4 shows the correlation between response variable and each predictor. Rooms has the highest positive correlation, and lstat has the highest negative correlation with the dependent variable. In other words, the higher the number of rooms and the lower the percentage of lower socio-economic population, the higher the home value. This concludes the EDA part. Regarding the models, Lasso regression and elastic net have low R^2 and high RMSE, so we recommend against these methods. Linear regression has higher R^2, but ridge regression has lower RMSE. Since RMSE is the index of prediction error in this study, we recommend ridge regression method as the best model. The intercept and coefficients in exhibit 5 from ridge regression can be used for management to build an equation that can forecast home value.

Appendix

Exhibit 1

```
General description of the boston_input DataFrame:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 506 entries, 0 to 505
Data columns (total 14 columns):
neighborhood    506 non-null object
crim            506 non-null float64
zn              506 non-null float64
indus           506 non-null float64
chas            506 non-null int64
nox             506 non-null float64
rooms           506 non-null float64
age             506 non-null float64
dis             506 non-null float64
rad             506 non-null int64
tax             506 non-null int64
ptratio         506 non-null float64
lstat           506 non-null float64
mv              506 non-null float64
dtypes: float64(10), int64(3), object(1)
memory usage: 55.4+ KB
None
```

Exhibit 2

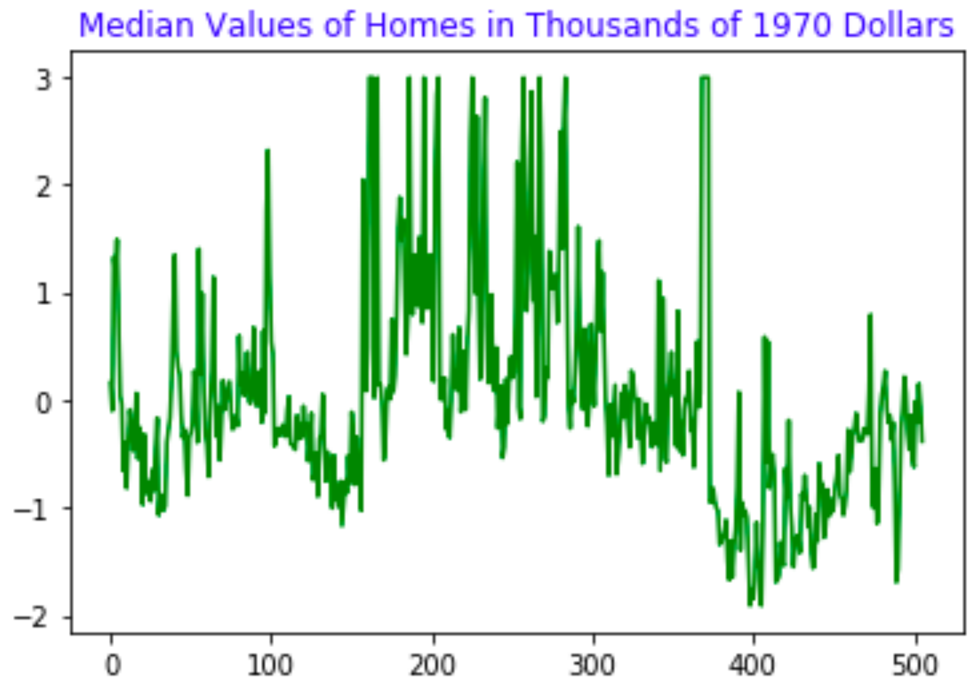
Descriptive statistics of the boston DataFrame:

	crim	zn	indus	chas	nox	rooms
count	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000
mean	3.613524	11.363636	11.136779	0.069170	0.554695	6.284634
std	8.601545	23.322453	6.860353	0.253994	0.115878	0.702617
min	0.006320	0.000000	0.460000	0.000000	0.385000	3.561000
25%	0.082045	0.000000	5.190000	0.000000	0.449000	5.885500
50%	0.256510	0.000000	9.690000	0.000000	0.538000	6.208500
75%	3.677082	12.500000	18.100000	0.000000	0.624000	6.623500
max	88.976200	100.000000	27.740000	1.000000	0.871000	8.780000

	age	dis	rad	tax	ptratio	lstat
count	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000
mean	68.574901	3.795043	9.549407	408.237154	18.455534	12.653063
std	28.148861	2.105710	8.707259	168.537116	2.164946	7.141062
min	2.900000	1.129600	1.000000	187.000000	12.600000	1.730000
25%	45.025000	2.100175	4.000000	279.000000	17.400000	6.950000
50%	77.500000	3.207450	5.000000	330.000000	19.050000	11.360000
75%	94.075000	5.188425	24.000000	666.000000	20.200000	16.955000
max	100.000000	12.126500	24.000000	711.000000	22.000000	37.970000

	mv
count	506.000000
mean	22.528854
std	9.182176
min	5.000000
25%	17.025000
50%	21.200000
75%	25.000000
max	50.000000

Exhibit 3


Median Values of Homes in Thousands of 1970 Dollars
3
2
1
0
−1
−2
0
100
200
300
400
500

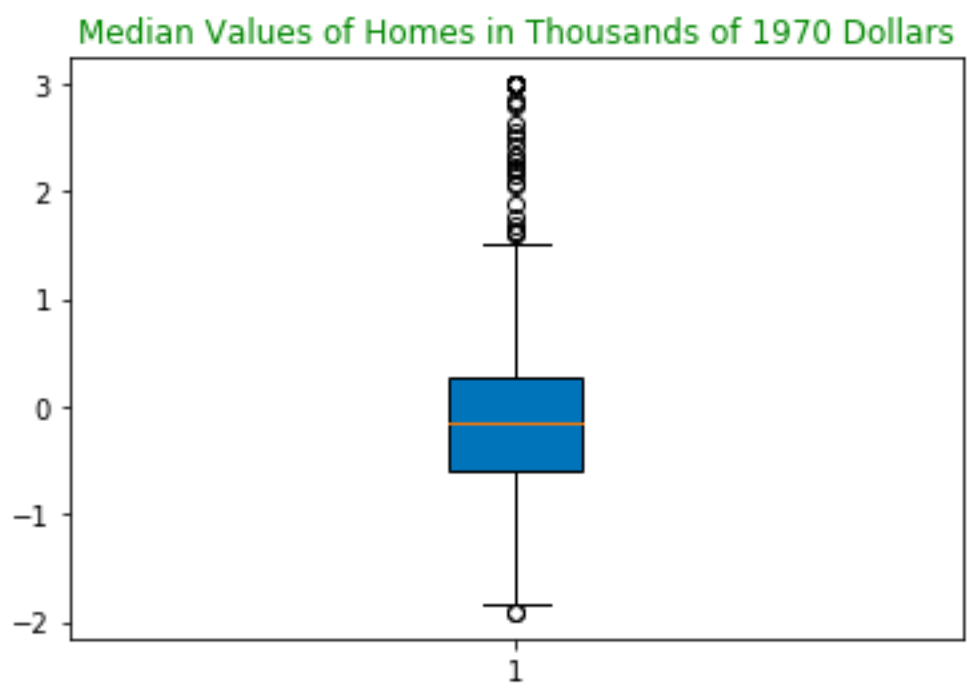
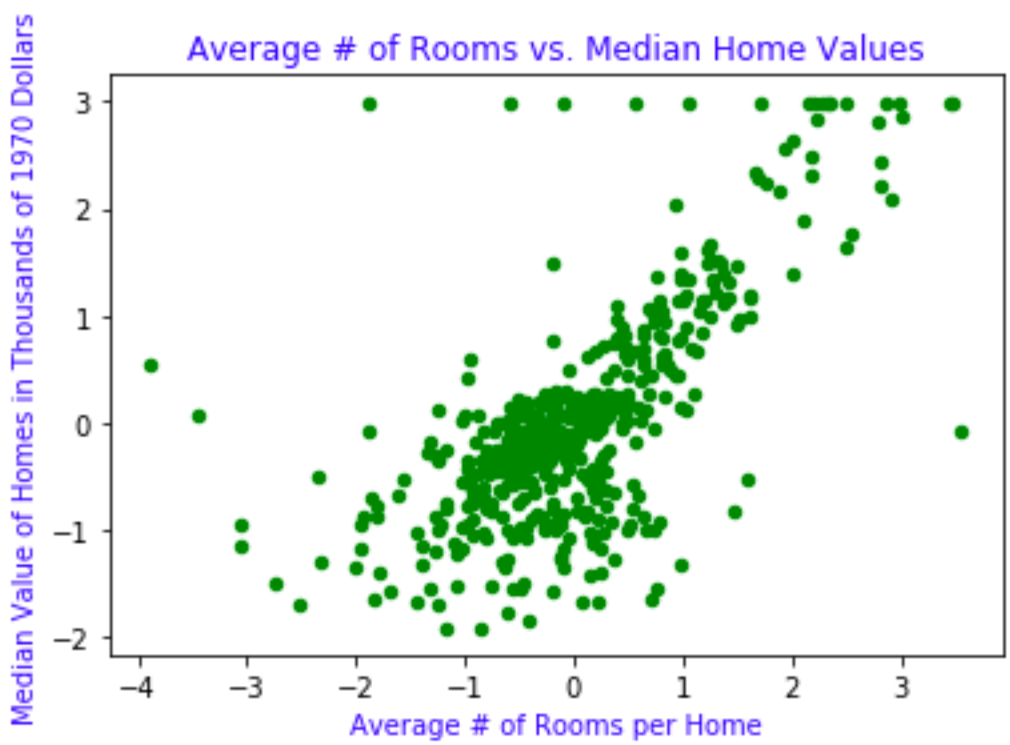

Median Values of Homes in Thousands of 1970 Dollars
3
2
1
0
−1
−2
1

Exhibit 4

```
mv          1.000000
rooms       0.696304
zn          0.360386
dis         0.249315
chas        0.175663
age        -0.377999
rad        -0.384766
crim       -0.389582
nox        -0.429300
tax        -0.471979
indus      -0.484754
ptratio    -0.505655
lstat      -0.740836
Name: mv, dtype: float64
```



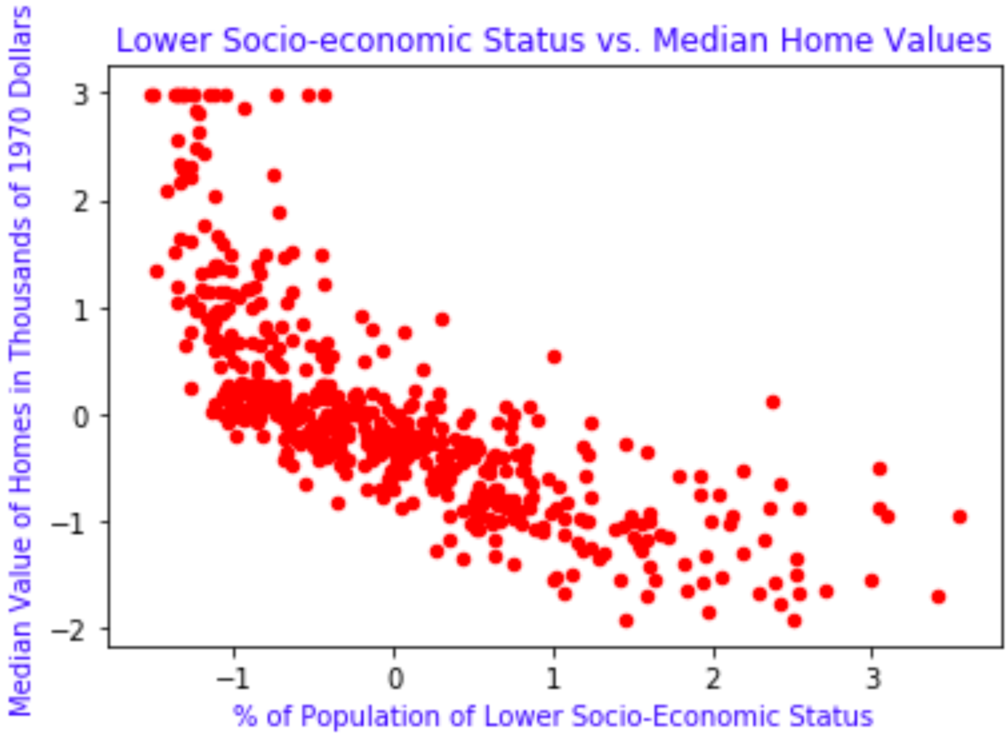


Exhibit 5

```
ridge.intercept_
# intercept of ridge regression
```

```
array([-7.72059356e-16])
```

```
ridge.coef_
# coefficients of ridge regression
```

```
array([[-0.1043611 ,  0.10798955, -0.00773088,  0.08141609, -0.2090253 ,
         0.28701971,  0.0039542 , -0.31983606,  0.21136091, -0.18369383,
        -0.20686589, -0.41898584]])
```