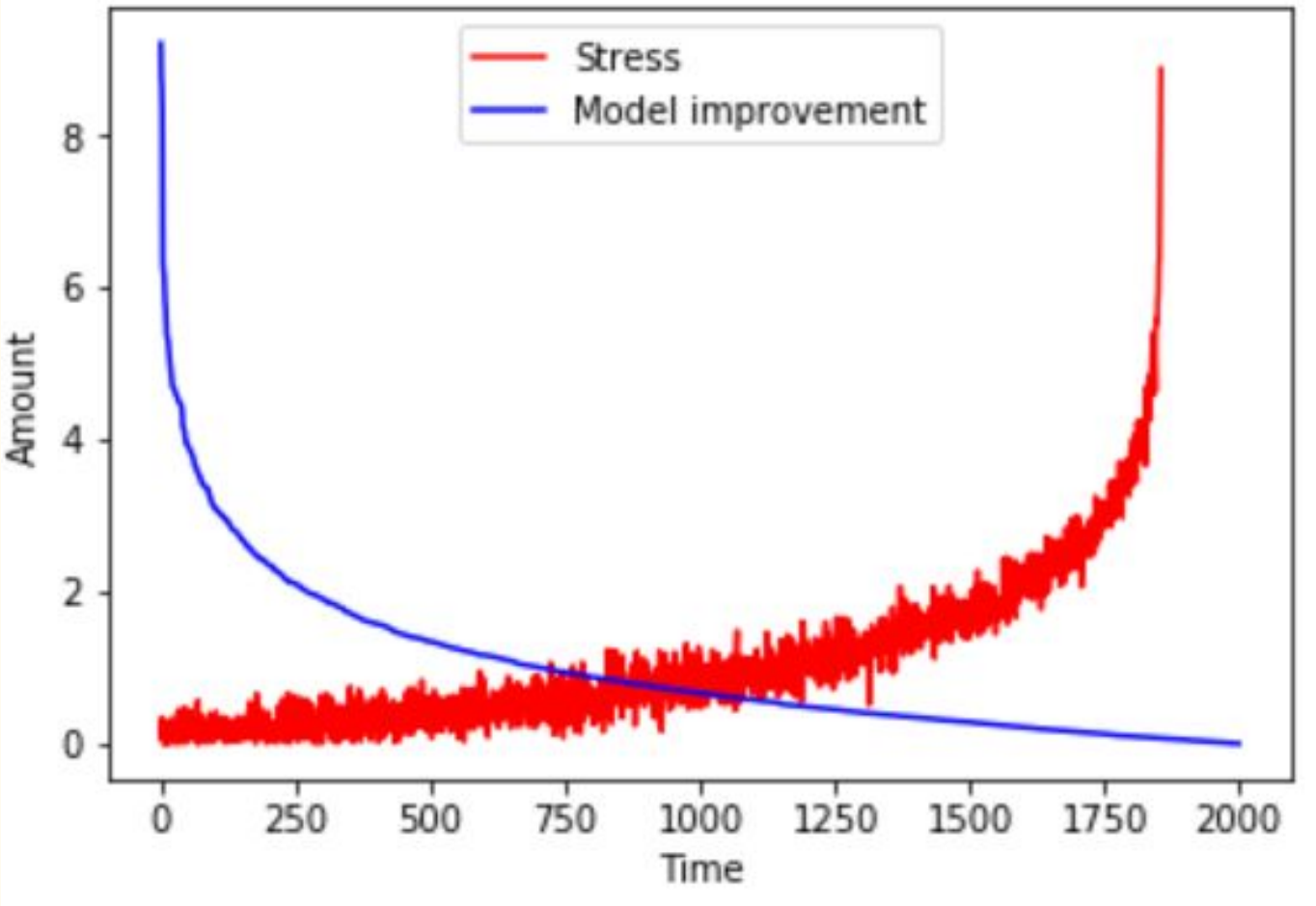
Ames Housing Dataset Predictions

—

Shawn Mitchell


Stress
Model improvement
Amount
Time
0
2
4
6
8
0
250
500
750
1000
1250
1500
1750
2000

Project Summary

Executive Summary:

Using the Ames housing dataset, a predictive linear regression model can be built to predict housing prices with reasonable accuracy. Lasso and ridge regularization methods are compared, with ridge performing the best.

Data Science Problem:

Is there are predictive relationship between the housing data features and the sale price?

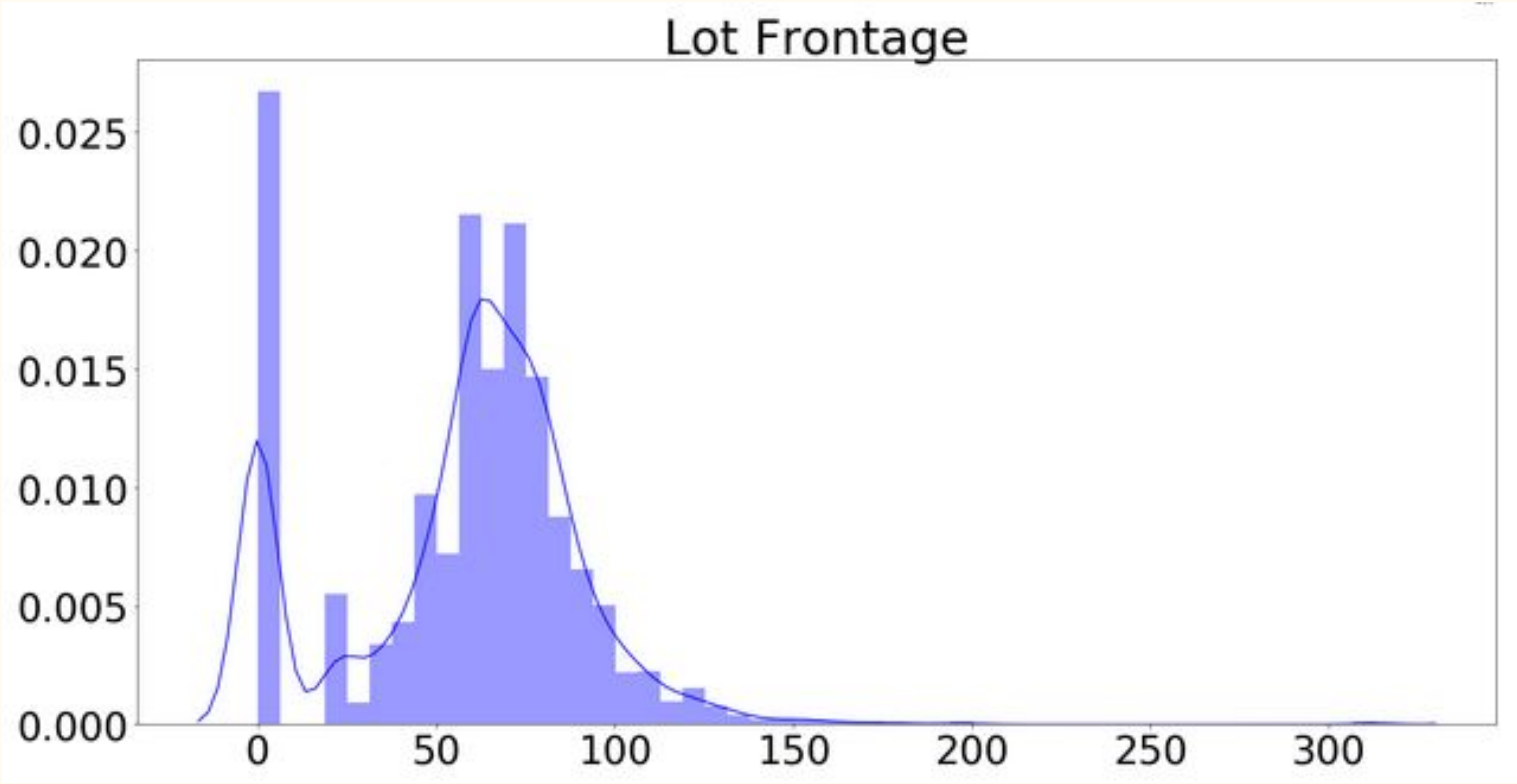
Initial Data Cleaning

1. Missing values
2. Clearly invalid values
3. Skewed data
4. Polynomial feature creation

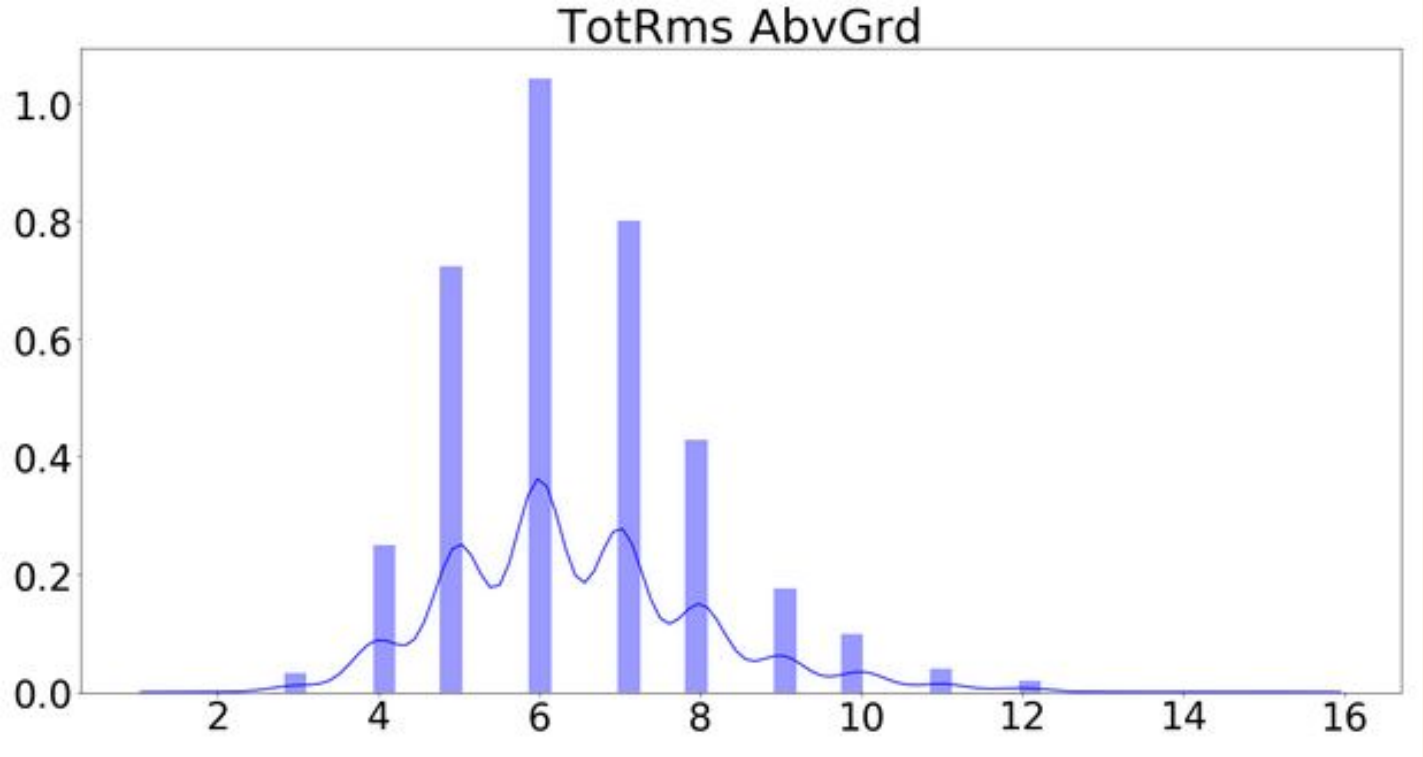
Correlations:

Id	Yr Sold	0.975747
Garage Cars	Garage Area	0.893442
Gr Liv Area	TotRms AbvGrd	0.813333
Total Bsmt SF	1st Flr SF	0.808351
Overall Qual	SalePrice	0.800207
Gr Liv Area	SalePrice	0.697038
Bedroom AbvGr	TotRms AbvGrd	0.655439
Garage Area	SalePrice	0.649897
Garage Cars	SalePrice	0.647781
BsmtFin SF 1	Bsmt Full Bath	0.645697
2nd Flr SF	Gr Liv Area	0.639092
Total Bsmt SF	SalePrice	0.629303
Year Built	Year Remod/Add	0.629116
1st Flr SF	SalePrice	0.618486
Gr Liv Area	Full Bath	0.617323

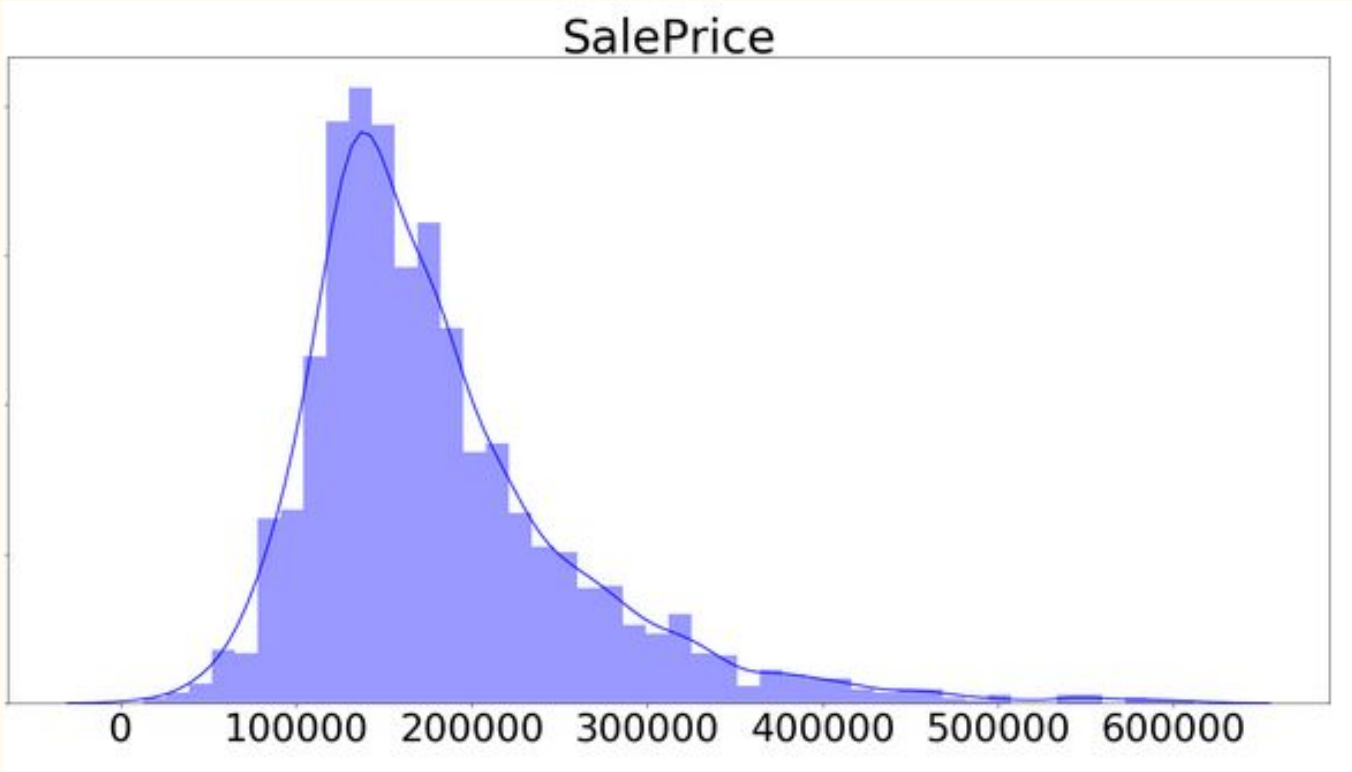
Skewed Data


Lot Frontage
0.025
0.020
0.015
0.010
0.005
0.000
0
50
100
150
200
250
300

Skewed Data



Skewed Data



Anomaly Detection

Using the Python library Luminol, we can see severity and frequency of outliers.

This ties in with skew of the data.

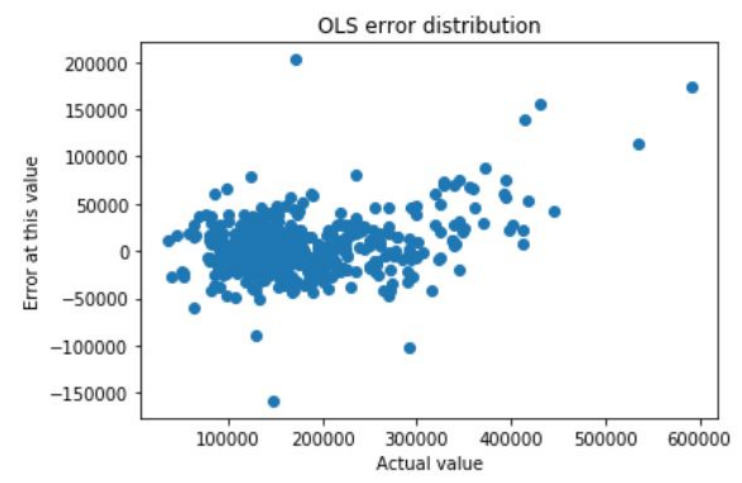
81: $502,000

1964: $592,000

```
81 3.643837953182729
138 3.929253129580728
151 3.8483737983628776
800 3.3408359469855355
823 3.9108297854559715
934 3.1633298399031986
1035 3.142850818270162
1117 3.12307370103948
1158 3.2025367338440125
1164 3.5908811587745473
1227 3.4036904502417373
1592 3.7126386455134353
1671 4.364757432424422
1692 4.164759332144492
1796 4.305198440600173
1964 4.1959486162680415
```

Final Data Cleaning

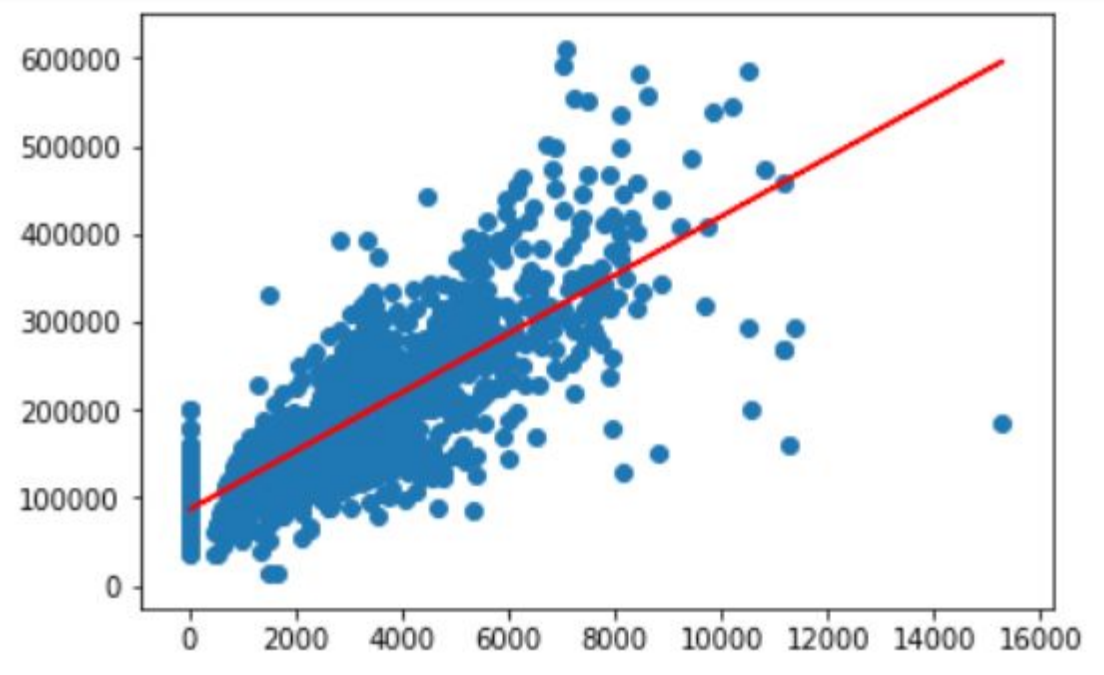
1. Basic model testing
2. Data exploration
3. Feature scaling
4. Categorical data replacement




```
OLS CV score: 0.4504400150770924
OLS train score: 0.8828247297577045
OLS test score: 0.8553666103256446
Ridge CV score: 0.540515742558779
Ridge train score: 0.881427528636412
Ridge test score: 0.8582167558632919
Lasso CV score: 0.466872351588569 16
Lasso train score: 0.7246841933887087
Lasso test score: 0.761818700889986
Elastic CV score: 0.2802235297909192
Elastic train score: 0.282362106896751 45
Elastic test score: 0.3062522505257077
```

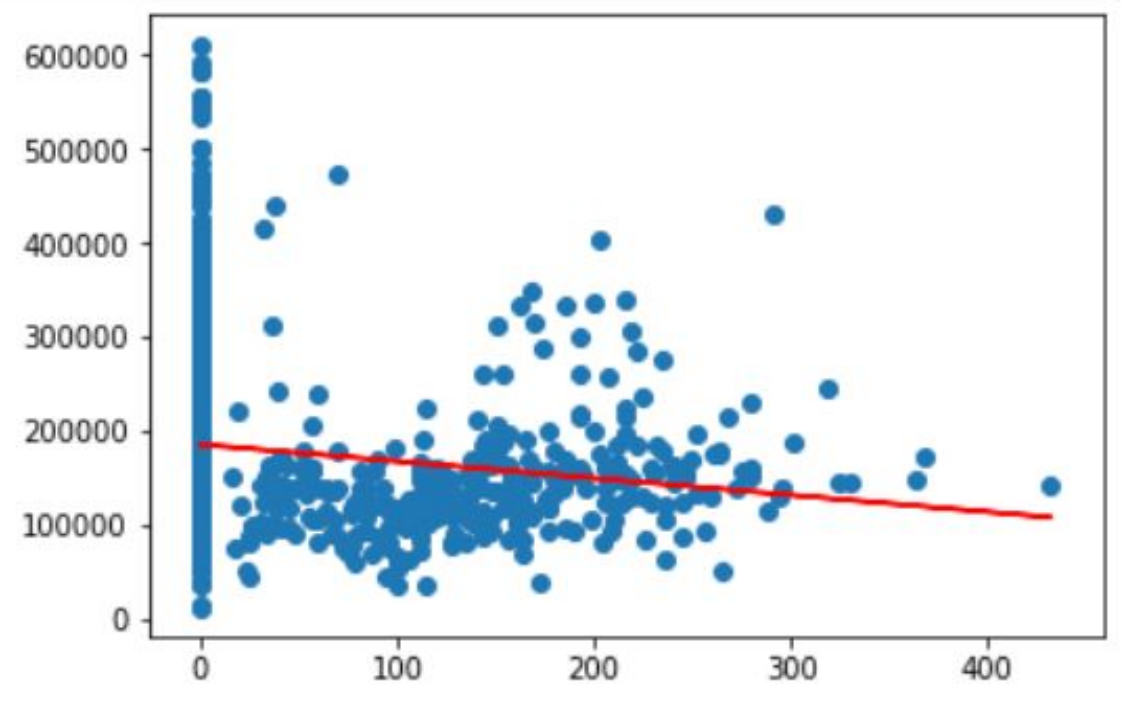
Correlations

'Garage Cars Gr Liv Area' (Polynomial)

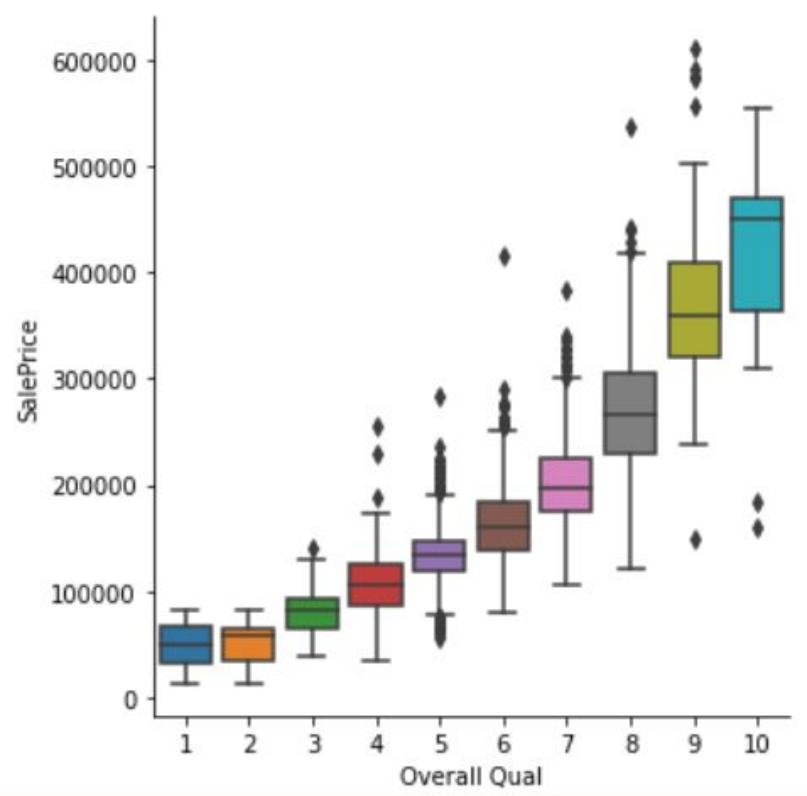


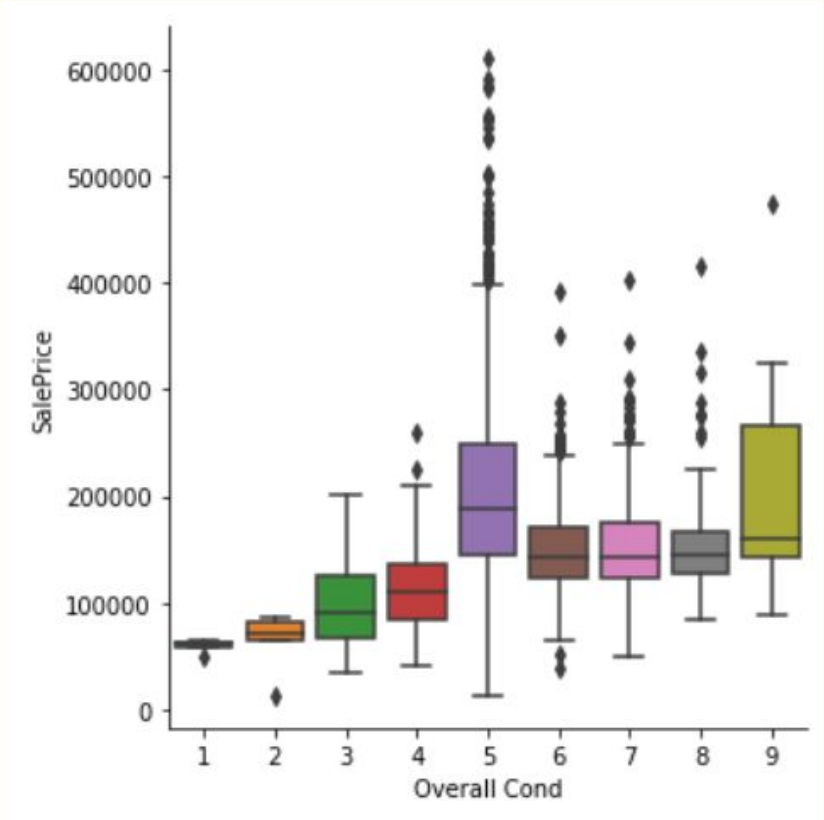
Correlation

'Enclosed Porch'

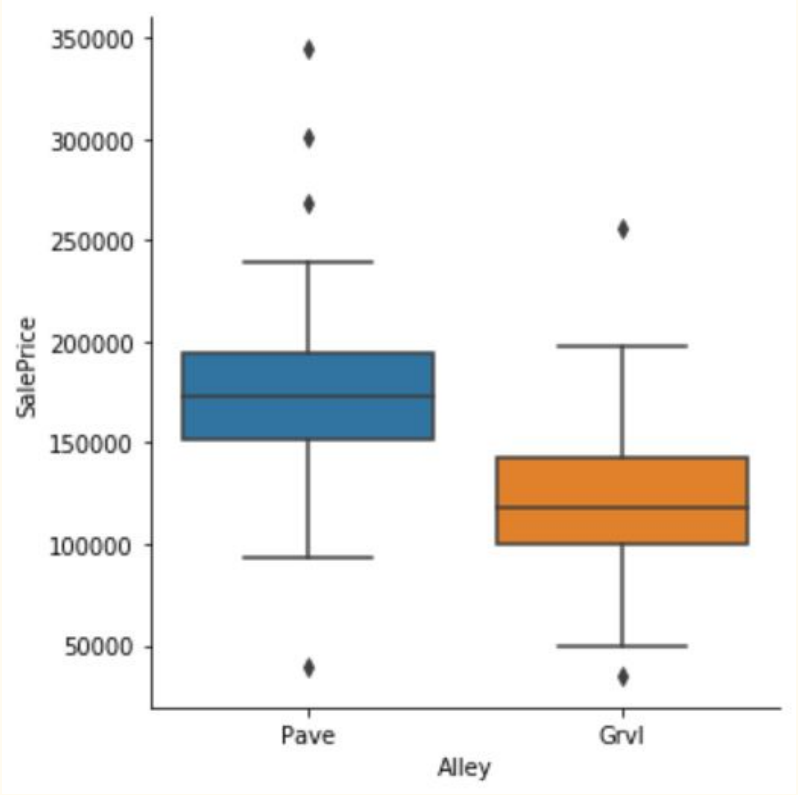


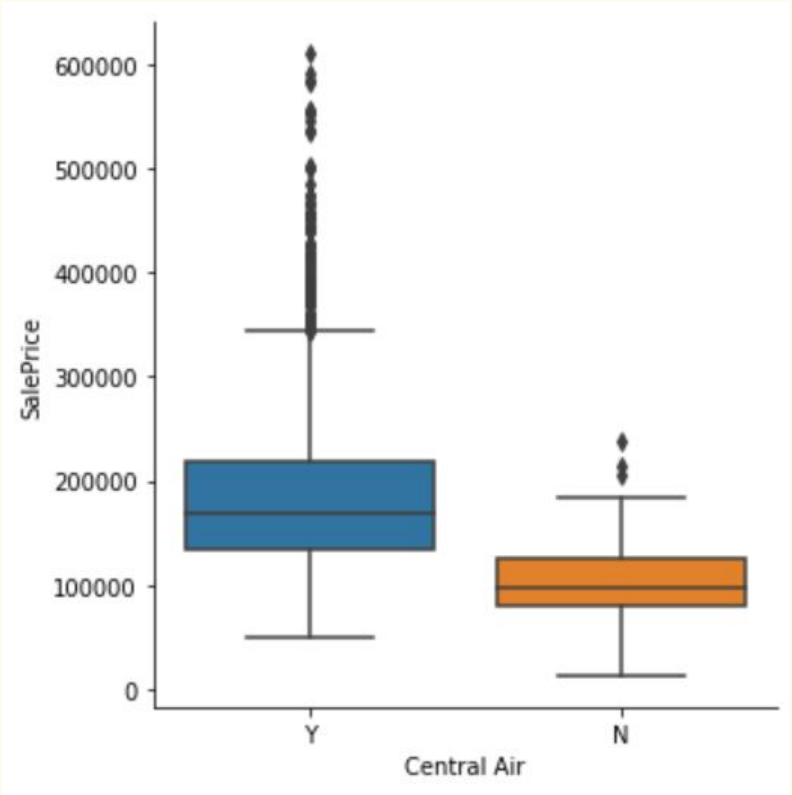
Categorical Pricing Ranges





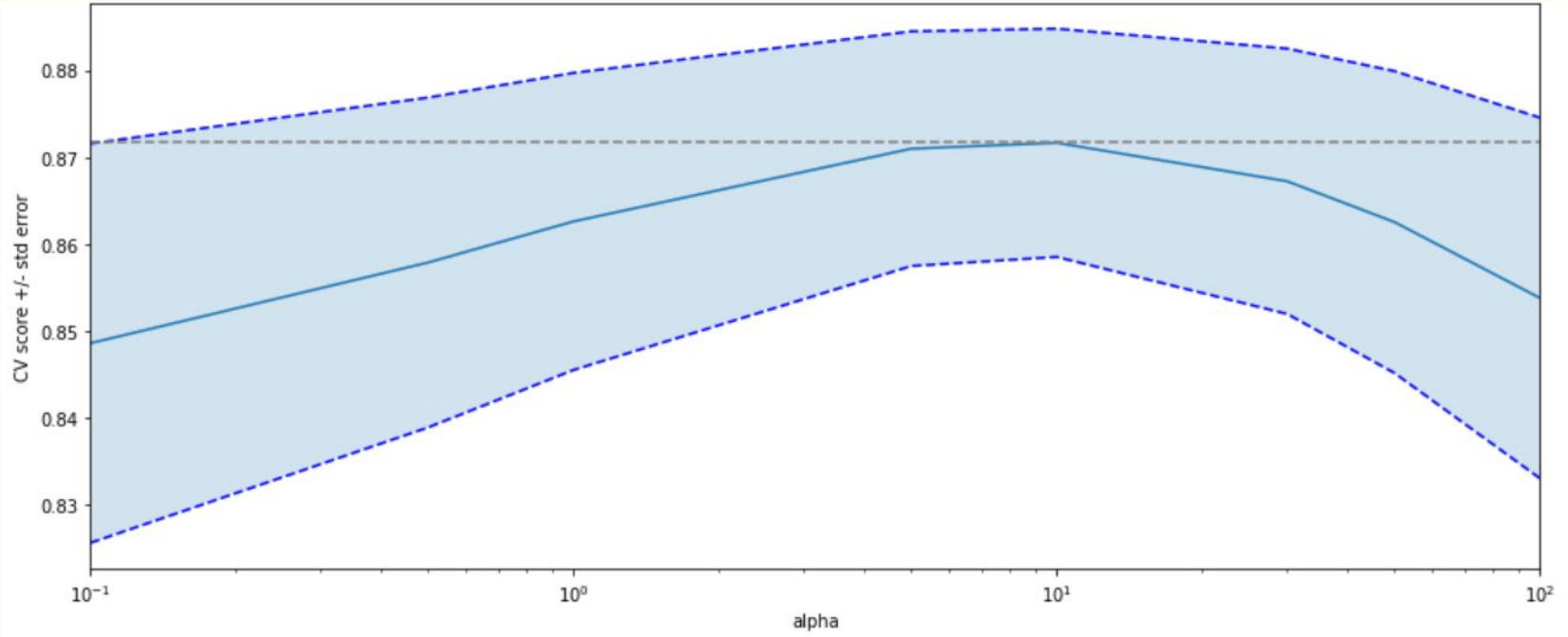
Categorical Pricing Ranges





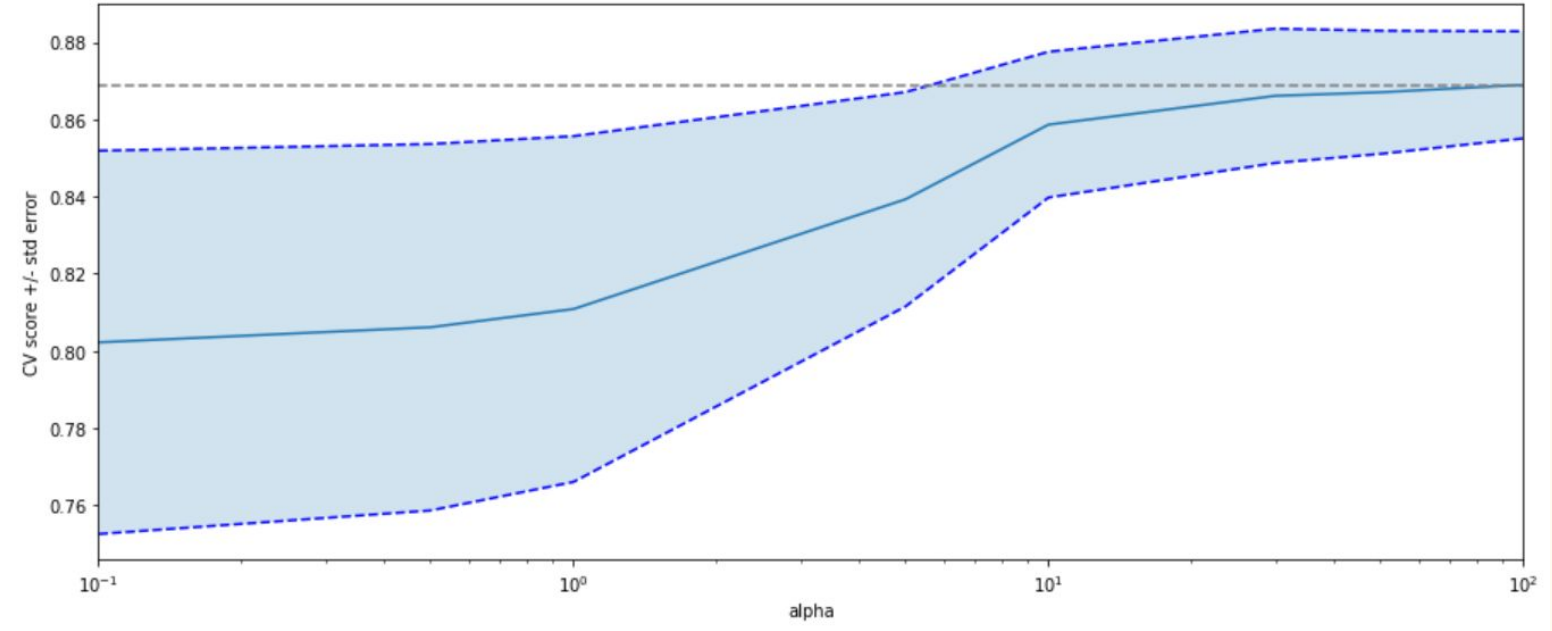
Hyperparameter Tuning

Ridge alphas: (0.1, 0.5, 1, 5, 10, 30, 50, 100)

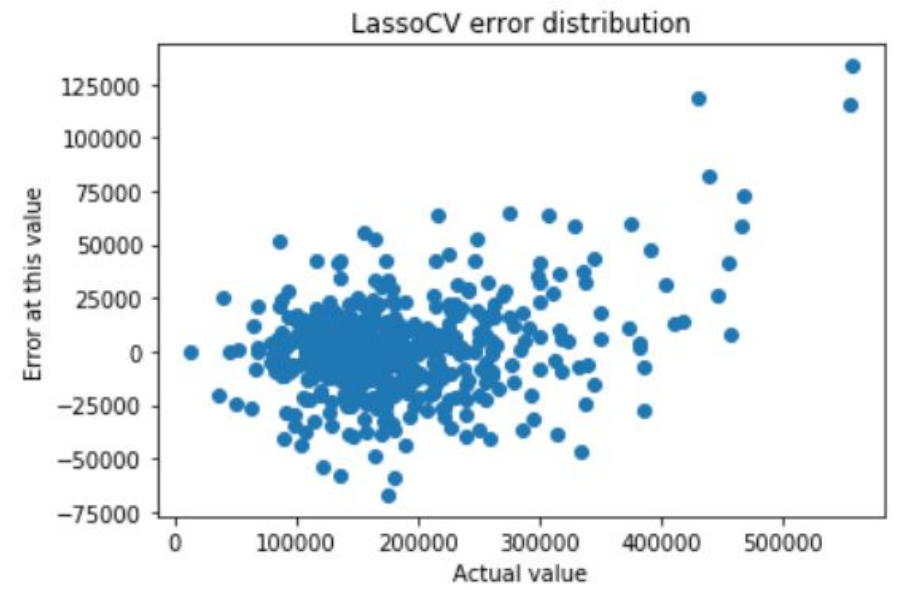


Hyperparameter Tuning

Lasso alphas: (0.1, 0.5, 1, 5, 10, 30, 50, 100)



LassoCV vs RidgeCV




```
lasso.score(X_train, y_train)
```

```
0.9258427607264067
```

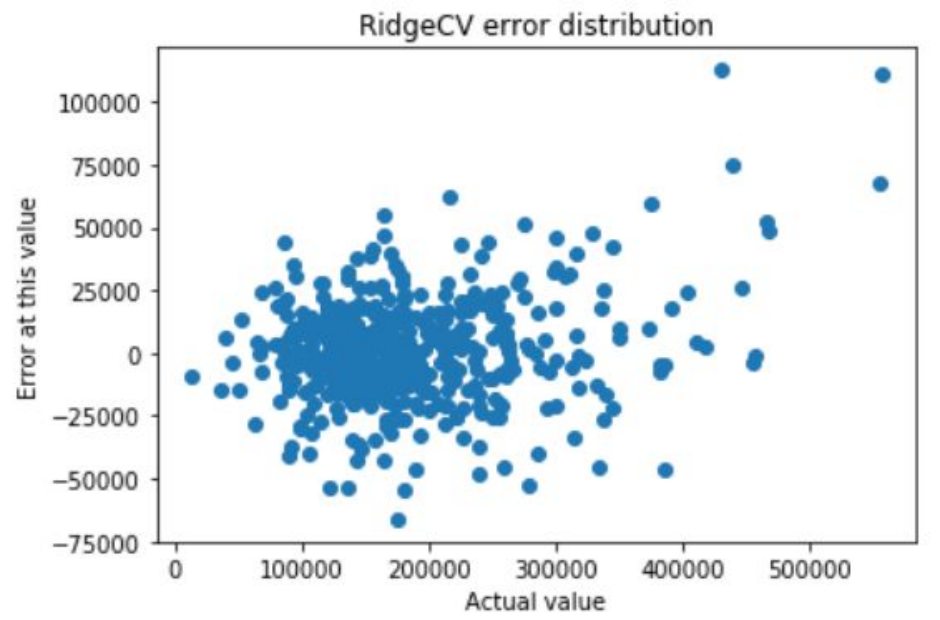
```
lasso.score(X_test, y_test)
```

```
0.9422364411570168
```

```
cross_val_score(lasso, X, y, cv=5).mean()
```

```
0.8628983746467658
```

LassoCV vs RidgeCV




```
ridge.score(X_train, y_train)
```

```
0.910739092910603
```

```
ridge.score(X_test, y_test)
```

```
0.9362450831112175
```

```
cross_val_score(ridge, X, y, cv=5).mean()
```

```
0.862544403884988
```

Final Production Model - RidgeCV

Kaggle score: 24,000

Recommendation: Using all features will produce a reasonably performing model, if regularized after.

Further improvement?

1. Feature removal
2. Polynomial feature review
3. Review of missing data decisions
4. Investigate improving high value home prices, without compromising low to mid value home predictions