Predicting Price using Linear Regression

Vilo Avila

Problem statement

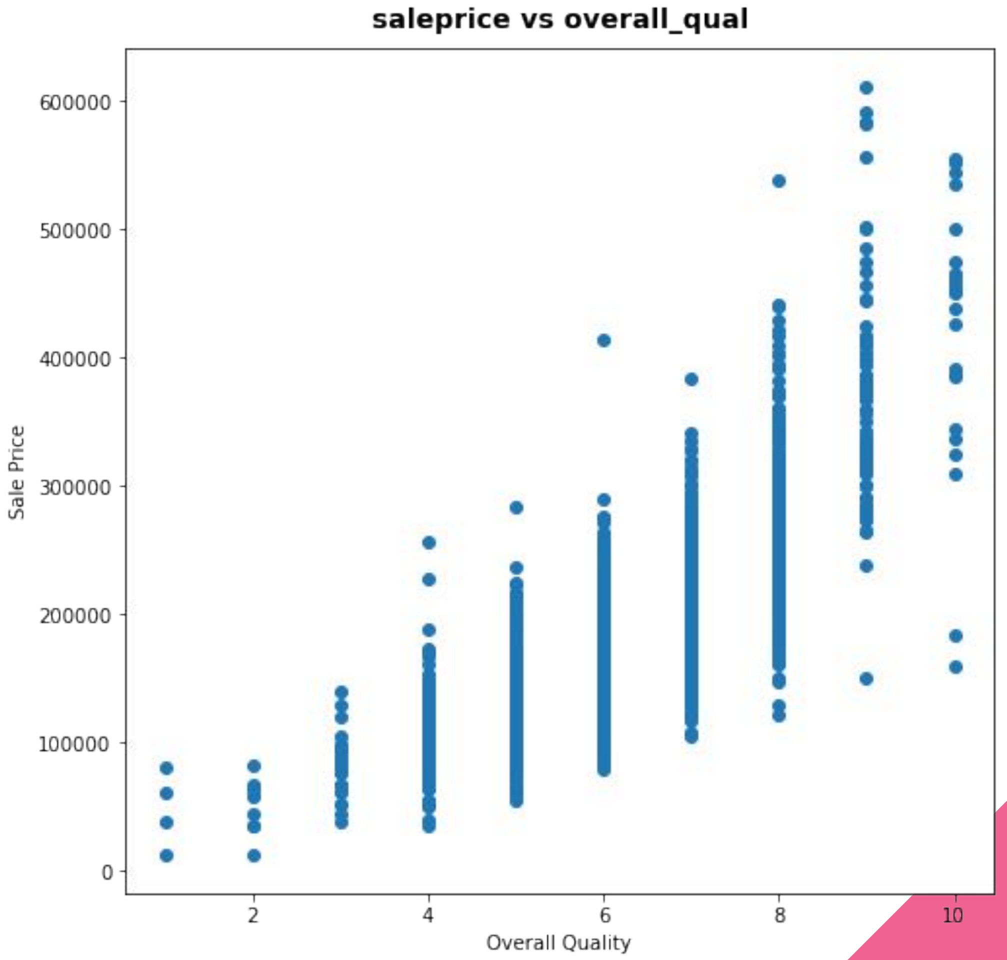
- We are tasked with predicting housing prices in a competition setting
- We are given a dataset to train our model in order to best predict pricing
- My goal is to discover what features would be best to use for my model

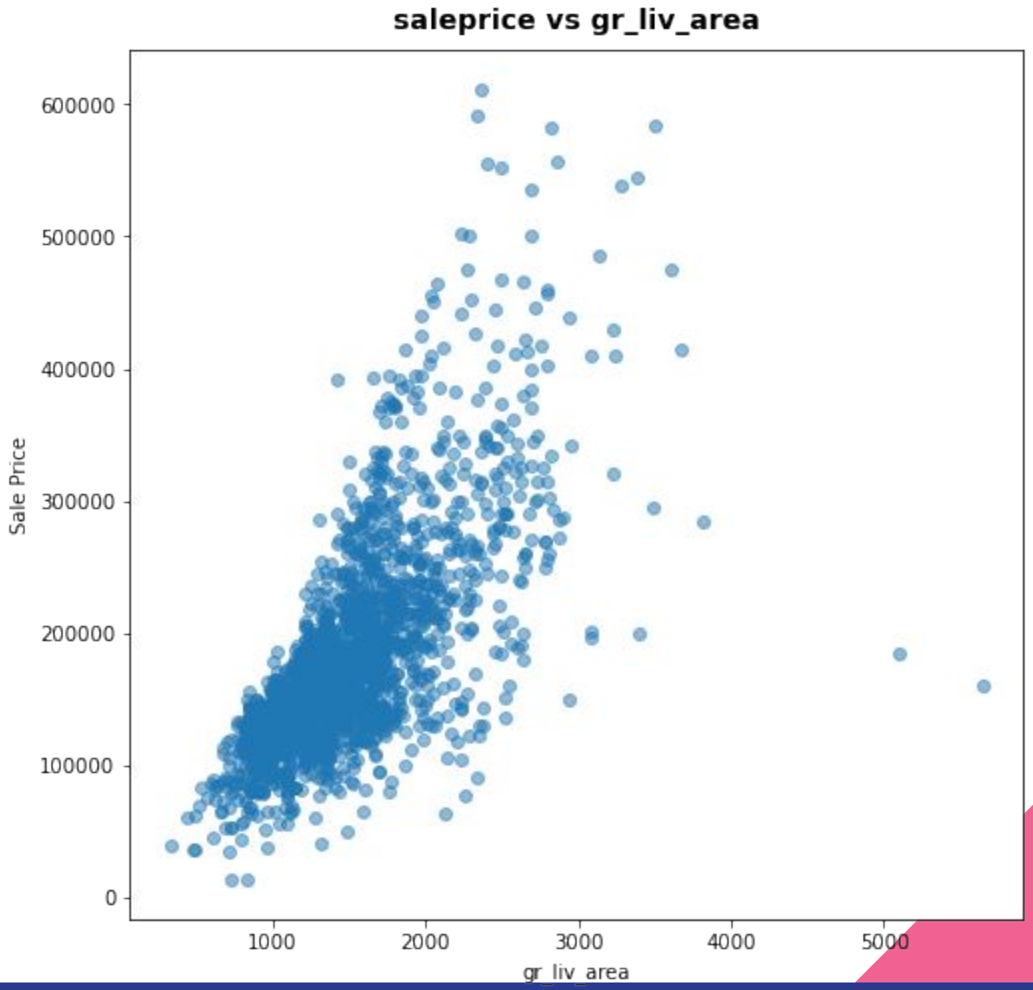
Methods used for cleaning training dataset

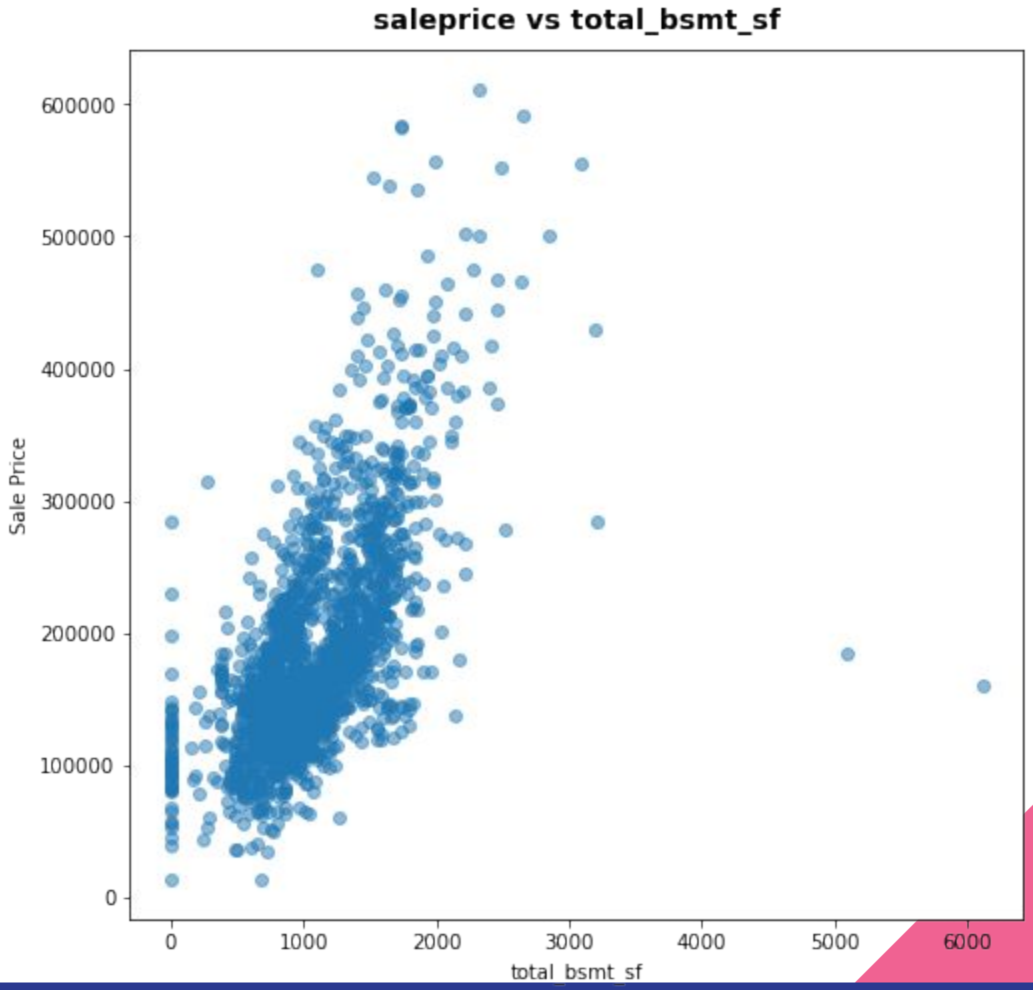
- First the columns were set to lower case and snake case
- All null values were checked
- Columns that had over 50% of null values were dropped, which totaled about 5 columns
- After dummies were added to dataframe the remaining null values were filled using mean of column
- Some outliers were removed after testing with models

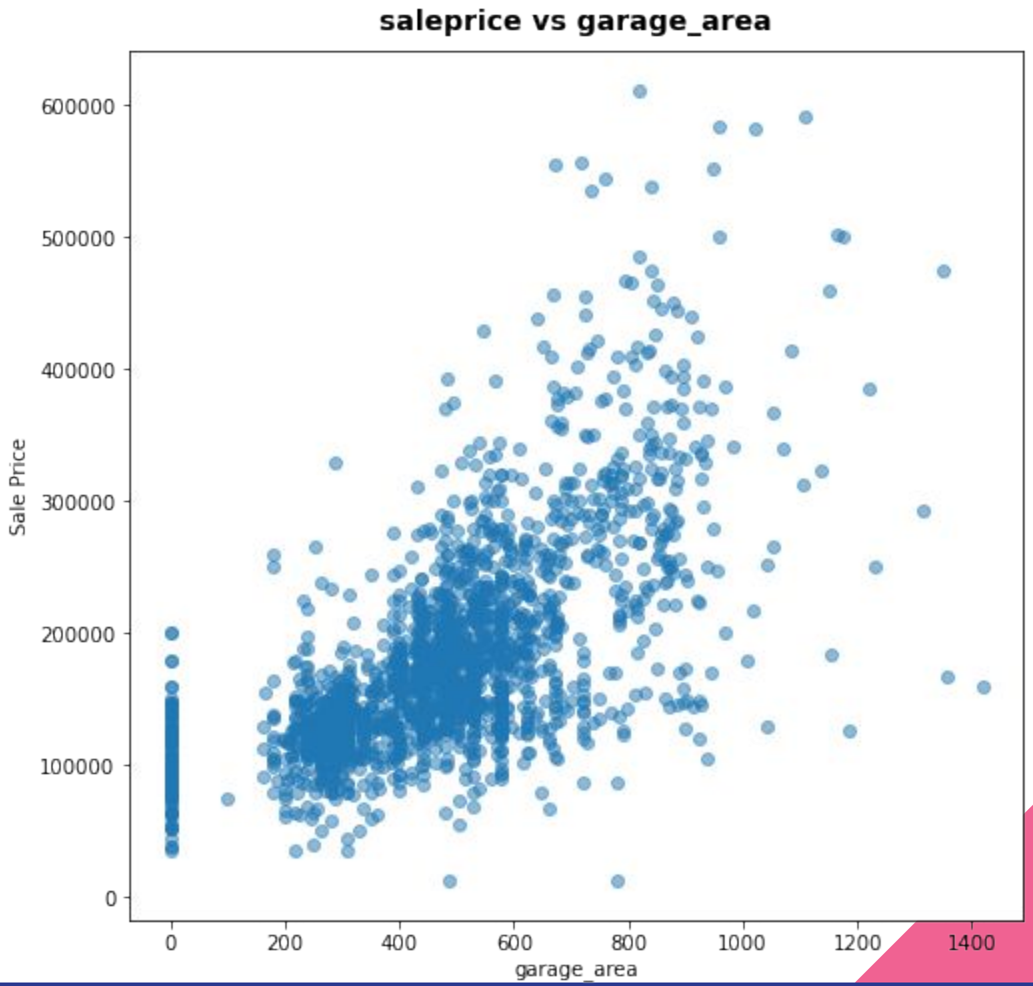
Looking at overall correlation between saleprice and numerical features using absolute values

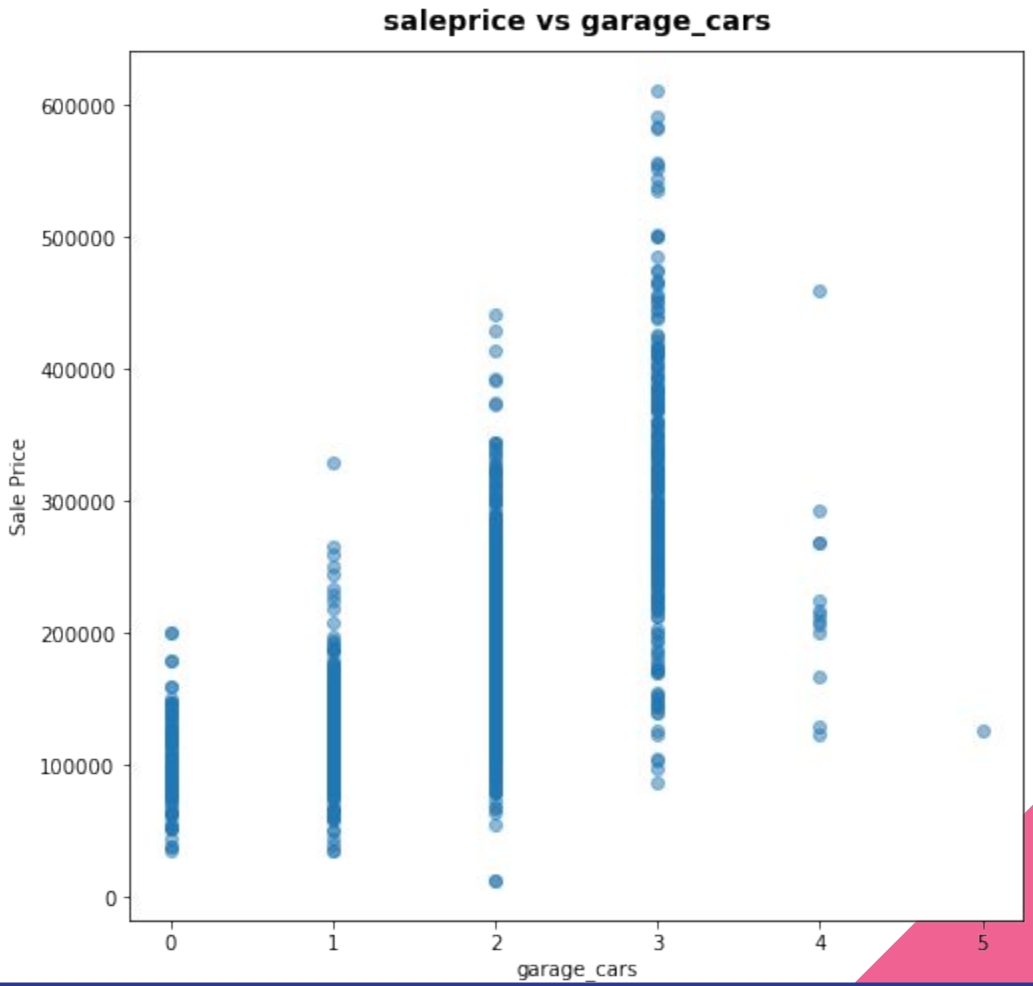
	saleprice
saleprice	1.000000
overall_qual	0.800207
gr_liv_area	0.697038
garage_area	0.650270
garage_cars	0.648220
total_bsmt_sf	0.628925
1st_flr_sf	0.618486
year_built	0.571849
year_remod/add	0.550370
full_bath	0.537969
garage_yr_blt	0.533922
mas_vnr_area	0.512230
totrms_abvgrd	0.504014
fireplaces	0.471093
bsmtfin_sf_1	0.423519
lot_frontage	0.341842
open_porch_sf	0.333476
wood_deck_sf	0.326490
lot_area	0.296566
bsmt_full_bath	0.283662


saleprice vs overall_qual
Sale Price
0
100000
200000
300000
400000
500000
600000
2
4
6
8
10
Overall Quality


saleprice vs gr_liv_area
600000
500000
400000
300000
200000
100000
0
Sale Price
1000
2000
3000
4000
5000
gr_liv_area


saleprice vs total_bsmt_sf
Sale Price
total_bsmt_sf
0
100000
200000
300000
400000
500000
600000
0
1000
2000
3000
4000
5000
6000


saleprice vs garage_area
Sale Price
600000
500000
400000
300000
200000
100000
0
0
200
400
600
800
1000
1200
1400
garage_area


saleprice vs garage_cars
Sale Price
600000
500000
400000
300000
200000
100000
0
0
1
2
3
4
5
garage_cars

	saleprice
saleprice	1.000000
overall_qual	0.808801
gr_liv_area	0.721258
total_bsmt_sf	0.663803
garage_area	0.658120
garage_cars	0.651989
1st_flr_sf	0.646387
exter_qual_TA	0.605444
bsmt_qual_Ex	0.591912
year_built	0.577410
kitchen_qual_Ex	0.553985
year_remod/add	0.553663
kitchen_qual_TA	0.542783
full_bath	0.538670
foundation_PConc	0.534196
mas_vnr_area	0.520154
garage_yr_blt	0.518424
totrms_abvgrd	0.508191
exter_qual_Ex	0.496969
fireplaces	0.470356
bsmtfin_type_1_GLQ	0.466772
bsmt_qual_TA	0.461304
heating_qc_Ex	0.458028
exter_qual_Gd	0.454933
neighborhood_NridgHt	0.444715
bsmtfin_sf_1	0.441580

Most Correlated After dummies

Most Correlated By Coefficient

	Feature	Coef
9	kitchen_qual_Ex	36650.971242
7	bsmt_qual_Ex	34906.046730
0	overall_qual	11651.885220
13	foundation_PConc	2833.198491
10	year_remod/add	305.312657
8	year_built	275.341936
1	gr_liv_area	59.603636
3	garage_area	42.193691
14	mas_vnr_area	32.716177
2	total_bsmt_sf	25.206856
5	1st_flr_sf	11.223018
15	garage_yr_blt	-87.758713
4	garage_cars	-1306.999494
16	totrms_abvgrd	-2217.126141
11	kitchen_qual_TA	-3618.666687
6	exter_qual_TA	-4494.919539
12	full_bath	-8416.352076

Cross Val Score

17 Total features selected out of 270 (after dummy values)

Min score: 0.86

Max score: 0.90

Mean Score 0.88

Confidence interval 0.03

RMSE - $27,065.76

Score: 31989.64376

Public score: 29157.63839

Other methods attempted

- Lasso
- LassoCV
- Ridge
- GridSearchCV
- SelectKBest

Conclusions

- I found that the best features to be used for the model were the ones I stated in previous slides
- More tweaking of features such as feature engineering and diving deeper into outliers could improve model
- More research into the different kinds of models to make better use of them for feature selection