Problem Statement -Part 2
Question 1
What is the optimal value of alpha for ridge and lasso regression? What will be the changes in the model if you choo se double the value of alpha for both ridge and lasso? What will be the most important predictor variables after the c hange is implemented?

Answer
The optimal value of alpha for ridge and lasso regression

Ridge Alpha 1

lasso Alpha 10

Ridge Regression

```
#Change the alpha value from 1 to 2
alpha = 3
ridge2 = Ridge(alpha=alpha)
ridge2.fit(X_train1, y_train)
Ridge(alpha=3)
# Lets calculate some metrics such as R2 score, RSS and RMSE
y_pred_train = ridge2.predict(X_train1)
y_pred_test = ridge2.predict(X_test1)

metric2 = []
r2_train_lr = r2_score(y_train, y_pred_train)
print(r2_train_lr)
metric2.append(r2_train_lr)

r2_test_lr = r2_score(y_test, y_pred_test)
print(r2_test_lr)
metric2.append(r2_test_lr)

rss1_lr = np.sum(np.square(y_train - y_pred_train))
print(rss1_lr)
metric2.append(rss1_lr)

rss2_lr = np.sum(np.square(y_test - y_pred_test))
print(rss2_lr)
metric2.append(rss2_lr)

mse_train_lr = mean_squared_error(y_train, y_pred_train)
print(mse_train_lr)
metric2.append(mse_train_lr**0.5)

mse_test_lr = mean_squared_error(y_test, y_pred_test)
print(mse_test_lr)
metric2.append(mse_test_lr**0.5)

#Alpha 1
#R2score(train) 0.884340040460635
#R2score(test)  0.869613280468847
0.8797315810932456
0.8710282148272899
607995142958.1411
```

320928407278.46216
680845624.8131479
729382743.8146868
R2score on training data has decreased but it has increased on testing data
Lasso

```
#Changed alpha 10 to 20
alpha =20
lasso20 = Lasso(alpha=alpha)
lasso20.fit(X_train1, y_train)
Lasso(alpha=20)
# Lets calculate some metrics such as R2 score, RSS and RMSE
y_pred_train = lasso20.predict(X_train1)
y_pred_test = lasso20.predict(X_test1)

metric3 = []
r2_train_lr = r2_score(y_train, y_pred_train)
print(r2_train_lr)
metric3.append(r2_train_lr)

r2_test_lr = r2_score(y_test, y_pred_test)
print(r2_test_lr)
metric3.append(r2_test_lr)

rss1_lr = np.sum(np.square(y_train - y_pred_train))
print(rss1_lr)
metric3.append(rss1_lr)

rss2_lr = np.sum(np.square(y_test - y_pred_test))
print(rss2_lr)
metric3.append(rss2_lr)

mse_train_lr = mean_squared_error(y_train, y_pred_train)
print(mse_train_lr)
metric3.append(mse_train_lr**0.5)

mse_test_lr = mean_squared_error(y_test, y_pred_test)
print(mse_test_lr)
metric3.append(mse_test_lr**0.5)

#R2score at alpha-10
#0.8859222400899005
#0.8646666084570094
```

0.8854019697956436
0.8670105921065014
579329522996.7144
330925704432.26794
648745266.5136778
752103873.7096999
R2score of training data has decrease and it has increase on testing data

```
#important predictor variables
betas = pd.DataFrame(index=X_train1.columns)
betas.rows = X_train1.columns
betas['Ridge2'] = ridge2.coef_
betas['Ridge'] = ridge.coef_
```

```
betas['Lasso'] = lasso.coef_
betas['Lasso20'] = lasso20.coef_
pd.set_option('display.max_rows', None)
betas.head(68)
```

LotArea---------------Lot size in square feet
OverallQual---------Rates the overall material and finish of the house
OverallCond--------Rates the overall condition of the house
YearBuilt-------------Original construction date
BsmtFinSF1--------Type 1 finished square feet
TotalBsmtSF------- Total square feet of basement area
GrLivArea-----------Above grade (ground) living area square feet
TotRmsAbvGrd----Total rooms above grade (does not include bathrooms)
Street_Pave--------Pave road access to property
RoofMatl_Metal----Roof material_Metal
Predictors are same but the coefficent of these predictor has changed

Question 2
You have determined the optimal value of lambda for ridge and lasso regression during the assignment. Now, which one will you choose to apply and why?

Answer:

The r2_score of lasso is slightly higher than lasso for the test dataset so we will choose lasso regression to solve this problem

Question 3
After building the model, you realised that the five most important predictor variables in the lasso model are not avai lable in the incoming data. You will now have to create another model excluding the five most important predictor v ariables. Which are the five most important predictor variables now?

```
X_train1
```

```
y_train
1108    181000
745     299800
1134    169000
512     129900
43      130250
33      165500
269     148000
789     187500
1038     97000
151     372402
344      85000
1218     80500
1040    155000
688     392000
1289    281000
1459    147500
1448    112000
733     131400
3       140000
123     153900
```

812	55993
1258	190000
929	222000
1348	215000
692	335000
1014	119200
412	222000
1425	142000
497	184000
603	151000
348	154000
481	374000
484	132500
1184	186700
353	105900
1415	175900
1000	82000
5	143000
112	383970
465	178740
859	250000
687	148800
1254	165400
783	165500
464	124000
1102	135000
1192	125000
677	109500
1193	165000
841	157500
252	173000
622	135000
711	102776
861	131500
604	221000
73	144900
926	285000
75	91000
1327	130500
234	216500
14	157000
686	227875
882	178000
331	139000
624	165150
578	146000
1033	230000
1312	302000
1087	252000
1392	123000
1337	52500
1383	112000
577	164500
1313	333168
1413	257000

1363	156932
1001	86000
302	205000
630	124000
397	169500
2	223500
6	307000
345	140200
821	93000
1439	197000
238	318000
1021	194000
30	40000
1019	213490
1074	194000
1309	179200
660	197900
1125	115000
742	179000
284	179200
28	207500
370	172400
54	130000
118	320000
1103	159500
62	202500
1290	180500
236	185500
133	220000
760	127500
646	98300
81	153500
1215	125000
970	135000
50	177000
1293	162900
573	170000
1136	119000
979	139000
1085	147000
584	133000
510	164900
715	165000
1247	169900
681	159434
67	226000
104	169500
878	148000
217	107000
309	360000
394	109000
568	316600
798	485000
674	140000
205	180500

122	136000
966	160000
1012	165000
227	106000
1166	245350
1106	179900
597	194201
874	66500
784	128000
747	265979
1324	147000
206	143900
152	190000
853	158000
90	109900
1188	195000
1080	145000
593	140000
900	110000
527	446261
183	200000
276	201000
340	202900
1042	196000
1050	176485
368	132000
109	190000
891	172500
825	385000
1097	170000
1034	119750
935	79900
1304	130000
1388	377500
1321	72500
334	192000
1454	185000
425	135000
260	176000
1107	274725
972	99500
338	202500
1401	193000
1187	262000
277	141000
1036	315500
1016	203000
80	193500
106	100000
767	160000
36	145000
1049	84900
491	133000
487	175000
1122	112000

790	160200
1382	157000
86	174000
95	185000
744	180000
92	163500
501	226700
866	248900
1008	240000
478	297000
1333	125500
1168	235000
442	162900
1272	137000
1311	203000
903	240000
116	139000
665	230500
726	222000
815	224900
69	225000
310	165600
892	154500
264	73000
247	140000
161	412500
1362	104900
409	339750
131	244000
1098	128000
278	415298
1406	133000
11	345000
492	172785
283	244600
648	155000
426	275000
1443	121000
1174	239000
609	118500
504	147000
1344	155835
160	162500
550	140000
1347	283463
591	451950
200	140000
1408	125500
823	139500
951	119900
788	107900
1452	145000
7	200000
920	201000
730	236500

1138	196000
392	106500
363	118000
1240	224900
1126	174000
1035	84000
286	159000
1151	149900
408	280000
1332	100000
810	181000
158	254900
575	118500
1112	129900
888	268000
1026	167500
1387	136000
482	155000
581	253293
996	136500
251	235000
927	176000
253	158000
1265	183900
235	89500
89	123600
164	152000
799	175000
734	108000
1133	239500
965	178900
795	171000
148	141000
1067	167900
446	190000
908	131000
556	141000
1157	230000
553	108000
914	173733
381	187750
423	315000
0	208500
395	129000
1379	167500
521	150000
1250	244000
1059	220000
1204	153500
1028	105000
836	153500
193	130000
493	155000
139	231500
862	152000

580	181900
781	175900
101	178000
1411	140000
204	110000
889	149500
453	210000
279	192000
77	127000
146	105000
332	284000
904	125500
360	156000
1427	140000
1077	138800
548	125000
898	611657
399	241000
544	179665
1359	315000
372	125000
462	62383
182	120000
31	149350
565	128000
216	210000
855	127000
1063	110500
190	315000
1207	200000
713	129000
194	127000
567	214000
1189	189000
959	155000
539	272000
82	245000
1222	143000
642	345000
1179	93000
718	341000
259	97000
800	200000
237	194500
180	177000
147	222500
1146	180000
1061	81000
1201	197900
208	277000
45	319900
757	158900
120	180000
42	144000
620	67000

875	303477
1455	175000
1404	105000
531	128000
513	134000
27	306000
890	122900
1109	280000
1027	293077
1235	138887
401	164990
554	284000
1284	169000
419	142000
802	189000
785	161500
1299	154000
498	130000
292	131000
1037	287000
257	220000
486	156000
432	122500
1139	144000
199	274900
21	139400
1279	68400
963	239000
1223	137900
930	201000
650	205950
68	80000
301	267000
1183	120000
1371	165500
307	89500
1320	156500
759	290000
403	258000
1343	177000
40	160000
803	582933
1177	115000
723	135000
980	178400
1421	127500
599	151000
1167	173000
753	275500
806	135500
214	161750
500	113000
430	85400
740	132000
830	166000

1436	120500
796	143500
460	263435
411	145000
450	110000
83	126500
1367	127000
1148	116900
549	263000
905	128000
1096	127000
672	165000
127	87000
458	161000
656	145500
659	167000
1399	137450
100	205000
932	320000
1394	246578
1398	138000
822	225000
547	129500
1156	179900
1162	129000
1160	146000
13	279500
433	181000
1029	118000
607	225000
46	239686
925	175000
186	173000
456	98000
628	135000
916	35311
295	142500
1104	106000
467	146500
755	172500
596	114504
1368	144000
739	190000
754	156000
1287	190000
936	184900
1429	182900
177	172500
1366	193000
285	164700
1141	197500
369	162000
1372	274300
18	159000
564	268000

358	130000
839	130500
663	137500
1285	132500
1197	144000
1007	88000
997	185000
245	241500
202	112000
1055	180000
111	180000
918	238000
826	109500
571	120000
832	237000
25	256300
391	215000
801	109900
1212	113000
1353	410000
667	193500
1242	170000
374	219500
473	440000
1068	151400
240	262500
1004	181000
226	290000
1270	260000
1191	174000
105	250000
1093	146000
76	135750
1274	139000
314	178000
761	100000
96	214000
476	208900
1084	187500
171	215000
84	168500
1268	381000
1457	266500
863	132500
632	82500
489	86000
770	134900
211	186000
300	157000
1227	147000
975	165000
261	276000
535	107500
1409	215000
611	148000

130	226000
695	176000
1314	119000
877	350000
819	224000
561	170000
690	141000
1072	91500
1239	265900
999	206000
1322	190000
140	115000
864	250580
909	174000
129	150000
518	211000
808	159950
1234	130000
1047	145000
64	219500
658	97500
860	189950
911	143500
132	150750
717	157000
1175	285000
732	222500
117	155000
693	108480
404	168000
296	152000
716	159500
899	135000
1032	310000
902	180000
957	132000
52	110000
383	76000
1259	151000
1089	197000
967	135000
845	171000
218	311500
114	259500
4	250000
675	148500
623	168500
59	124900
291	135900
1308	147000
1331	132500
485	147000
1101	119500
1217	229456
856	147000

919	176500
1303	232000
756	212000
337	214000
375	61000
1297	140000
347	157500
166	190000
138	230000
1435	174000
626	139900
762	215200
725	120500
57	196500
1147	174500
794	194500
915	75000
515	402861
1069	135000
714	130500
436	116000
560	121500
354	140000
22	230000
749	98000
38	109000
994	337500
907	250000
1310	335000
445	127500
1015	227000
588	143000
827	189000
273	139000
858	152000
537	111250
172	239000
1058	335000
938	239799
682	173000
750	96500
621	240000
1390	235000
20	325300
141	260000
1022	87000
1078	155900
1432	64500
213	156000
1091	160000
1426	271000
662	110000
203	149000
333	207000
545	229000

1360	189000
1224	184000
1210	189000
566	325000
522	159000
175	243000
19	139000
598	217500
719	128500
1200	116050
1118	140000
1225	145000
1056	185850
1374	250000
312	119900
872	116000
250	76500
29	68500
479	89471
1420	179900
254	145000
41	170000
818	155000
1144	80000
514	96500
1248	129500
1434	160000
752	217000
1341	155000
697	123500
1263	180500
1221	134000
1325	55000
876	132250
178	501837
1053	144500
901	153000
1241	248328
968	37900
1006	163500
169	228000
1335	167900
558	175000
1226	214000
115	176000
641	226000
1294	115000
60	158000
168	183500
440	555000
230	148000
1380	58500
10	129500
1113	134500
1159	185000

496	430000
281	185000
988	195000
1277	197900
971	173000
1378	83000
680	143000
379	179000
1255	127500
290	233230
167	325624
517	265000
698	138500
1445	129000
1375	239000
602	220000
1023	191000
536	188000
1318	275000
162	220000
439	110000
1051	200141
1370	105000
720	275000
508	161000
209	145000
664	423000
159	320000
820	183000
1079	126000
220	204900
974	167500
308	82500
324	242000
1326	79000
437	119000
135	174000
225	112000
917	135000
274	124500
1419	223000
1373	466500
1261	128900
421	215000
939	244400
242	79000
699	196000
1	181500
124	181000
763	337000
1395	281213
1048	115000
1286	143000
1316	295493
483	164000

1243	465000
724	320000
1334	125000
351	190000
709	109900
776	221500
1070	135000
684	221000
689	194700
1424	144000
1346	262500
870	109500
704	213000
1111	205000
608	359100
746	236000
629	168500
1213	145000
1295	138500
574	139000
1158	235128
592	138000
1307	138000
817	271000
1351	171000
532	107500
107	115000
1119	133700
615	137500
811	144500
952	133900
982	159895
1137	94000
229	192500
429	175000
869	236000
961	272000
434	81000
837	100000
1339	128500
1127	259000
700	312500
668	168000
1057	248000
793	225000
748	260400
110	136900
880	157000
444	210000
551	112500
1172	171900
1155	218000
1024	287000
852	164000
223	97000

149	115000
1150	124000
44	141000
1116	184100
255	230000
414	228000
318	260000
459	110000
428	195400
647	155000
1271	185750
1190	168000
708	179540
1260	181000
569	135960
71	129500
16	149000
1291	119500
1164	194000
540	315000
415	181134
210	98000
834	139950
239	113000
865	148500
1178	154900
969	140000
371	134432
787	233000
326	324000
1115	318000
499	120000
893	165000
298	175000
1094	129000
224	386250
179	100000
797	110000
438	90350
768	216837
1236	175500
617	105500
1120	118400
1124	163900
1208	140000
228	125000
37	153000
847	133500
557	108000
1437	394617
103	198900
989	197000
616	183200
849	187000
268	120500

```
15      132000
349     437154
655      88000
424     139000
184     127000
923     193000
23      129900
707     254000
1262    161500
912      88000
1385    125500
614      75500
1423    274970
751     162000
1054    255000
212     252678
1073    159500
765     264132
559     234000
1185    104900
519     234000
944     137500
816     137000
280     228500
24      154000
503     289000
1296    155000
305     305900
1052    165000
1329    176500
422     113000
881     187500
377     340000
840     140000
145     130000
962     155000
1228    367294
98       83000
364     190000
1355    170000
511     202665
134     180000
1143     80000
1199    148000
1237    195000
1418    124000
949     197500
1233    142000
Name: SalePrice, dtype: int64
X_train1.columns
Index(['LotArea', 'OverallQual', 'OverallCond', 'YearBuilt', 'BsmtFinSF1', 'TotalBsmtSF', '1stFlrSF', '2ndFlrSF', 'GrLivArea', 'BedroomAbvGr', 'TotRmsAbvGrd', 'Street_Pave', 'LandSlope_Sev', 'Condition2_PosN', 'RoofStyle_Shed', 'RoofMatl_Metal', 'Exterior1st_Stone', 'Exterior2nd_CBlock', 'ExterQual_Gd', 'ExterQual_TA', 'BsmtCond_Po', 'KitchenQual_TA', 'Functional_Maj2', 'SaleType_CWD', 'SaleType_Con'], dtype='object')
```

LotArea,OverallQual,YearBuilt,BsmtFinSF1,TotalBsmtSF are the top 5 important predictors.

Let's drop these columns

```
X_train2 = X_train1.drop(['LotArea','OverallQual','YearBuilt','BsmtFinSF1','TotalBsmtSF'],axis=1)
X_test2 = X_test1.drop(['LotArea','OverallQual','YearBuilt','BsmtFinSF1','TotalBsmtSF'],axis=1)
X_train2.head()

X_test2.head()
```

Lasso

```
# alpha 10
alpha =10
lasso21 = Lasso(alpha=alpha)
lasso21.fit(X_train2, y_train)
Lasso(alpha=10)
# Lets calculate some metrics such as R2 score, RSS and RMSE
y_pred_train = lasso21.predict(X_train2)
y_pred_test = lasso21.predict(X_test2)

metric3 = []
r2_train_lr = r2_score(y_train, y_pred_train)
print(r2_train_lr)
metric3.append(r2_train_lr)

r2_test_lr = r2_score(y_test, y_pred_test)
print(r2_test_lr)
metric3.append(r2_test_lr)

rss1_lr = np.sum(np.square(y_train - y_pred_train))
print(rss1_lr)
metric3.append(rss1_lr)

rss2_lr = np.sum(np.square(y_test - y_pred_test))
print(rss2_lr)
metric3.append(rss2_lr)

mse_train_lr = mean_squared_error(y_train, y_pred_train)
print(mse_train_lr)
metric3.append(mse_train_lr**0.5)

mse_test_lr = mean_squared_error(y_test, y_pred_test)
print(mse_test_lr)
metric3.append(mse_test_lr**0.5)
#R2score at alpha-10
#0.8859222400899005
#0.8646666084570094
0.7988346707068132
0.758810320925813
1016954777102.8657
600167078819.8159
1138807141.2126155
1364016088.2268543
```

R2score of training and testing data has decreased

```
#important predictor variables
```

```
betas = pd.DataFrame(index=X_train2.columns)
betas.rows = X_train1.columns
betas['Lasso21'] = lasso21.coef_
pd.set_option('display.max_rows', None)
betas.head(68)
```

five most important predictor variables

11stFlrSF-----------First Floor square feet
GrLivArea-----------Above grade (ground) living area square feet
Street_Pave---------Pave road access to property
RoofMatl_Metal------Roof material_Metal
RoofStyle_Shed------Type of roof(Shed)
Question 4
How can you make sure that a model is robust and generalisable? What are the implications of the same for the accu racy of the model and why?

Answer

The model should be generalized so that the test accuracy is not lesser than the training score. The model should be a ccurate for datasets other than the ones which were used during training. Too much importance should not given to t he outliers so that the accuracy predicted by the model is high. To ensure that this is not the case, the outliers analysi s needs to be done and only those which are relevant to the dataset need to be retained. Those outliers which it does not make sense to keep must be removed from the dataset. If the model is not robust, It cannot be trusted for predicti ve analysis.