Question 1

What is the optimal value of alpha for ridge and lasso regression? What will be the changes in the model if you choose double the value of alpha for both ridge and lasso? What will be the most important predictor variables after the change is implemented?

Answer:

The optimal value of Alpha for Ridge regression is **3.5** and for Lasso regression **0.001.**

Below are the Metrics before and After doubled the Alpha.

	Metric Name	Ridge	Lasso	Ridge_DoubleAlpha	Lasso_DoubleAlpha
0	R2-Score_Train	0.940999	0.935200	0.936717	0.927944
1	MSE_Train	0.059001	0.064800	0.063283	0.072056
2	RMSE_Train	0.242902	0.254557	0.251561	0.268433
3	R2-Score_Test	0.915457	0.920167	0.917978	0.920668
4	MSE_Test	0.082951	0.078329	0.080478	0.077838
5	RMSE_Test	0.288012	0.279874	0.283686	0.278995

- Ridge Model R2-Score (Train) decreased from 0.940999 to 0.936717.
- Lasso Model R2-Score(Train) decreased from 0.935200 to 0.927944.
- Ridge Model R2-Score(Test) increased slightly from 0.915457 to 0.917978.
- Lasso Model R2-Score(Test) increased slightly from 0.920167 to 0.920668.

Below are the Top 10 Predictor variables in Lasso model after doubled the Alpha.

	features	coefficient
120	OverallQual_10	1.340337
119	OverallQual_9	0.613546
10	GrLivArea	0.361475
265	SaleType_New	0.295306
69	Neighborhood_Crawfor	0.268876
85	Neighborhood_StoneBr	0.250009
118	OverallQual_8	0.230934
226	Functional_Typ	0.219956
139	Exterior1st_BrkFace	0.208813
78	Neighborhood_NoRidge	0.190664

- One predictor variable **Functional_Typ** got included in Top 10 after doubled the alpha and removed **OverAllCond_9.**

Below are Top 10 predictors after double the alpha in Ridge model.

	features	coefficient
120	OverallQual_10	0.519910
85	Neighborhood_StoneBr	0.283701
128	OverallCond_9	0.271303
119	OverallQual_9	0.236397
78	Neighborhood_NoRidge	0.223703
226	Functional_Typ	0.214295
10	GrLivArea	0.205905
69	Neighborhood_Crawfor	0.185729
139	Exterior1st_BrkFace	0.184958
189	BsmtExposure_Gd	0.183715

- For Ridge model the Top 10 predictors are same though co-efficient and order got 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:

- I would choose Lasso model as it eliminated some of the features as co-efficient are zero (highlighted in yellow) in below equation and still accuracy is more than Ridge for test dataset.

```
Lasso House Prediction=-
0.3039883160492747З+(1.4497736091906284*OverallQual_10)+(0.658890176343473
2*OverallQual_9)+(0.3464558544577409*GrLivArea)+(0.33955954590658805*Neigh
borhood_StoneBr)+(0.3078520306337739*SaleType_New)+(0.2766770588161336*Ove
rallQual_8)+(0.2748489292829179*Neighborhood_Crawfor)+(0.2663305641682858*
OverallCond_9)+(0.26261616452184167*Neighborhood_NoRidge)+(0.2372728445669
4792*Exterior1st_BrkFace)+(0.21776322857149405*Neighborhood_NridgHt)+(0.20
97726192201768*Functional_Typ)+(0.19289368041530794*BsmtExposure_Gd)+(0.11
724815520485306*Neighborhood_BrkSide)+(0.10823915253257556*TotalBsmtSF)+(0
.10613638360667003*Condition1_Norm)+(0.1049450121410171*OverallCond_7)+(0.
09962380273957354*BsmtFinSF1)+(0.09713993920757476*MSZoning_FV)+(0.0877440
8777439793*OverallCond_8)+(0.08226251836274859*OverallQual_7)+(0.072670166
85508504*SaleCondition_Normal)+(0.06991100248773673*HouseStyle_1.5Unf)+(0.
06568993111645202*BsmtFinType1_GLQ)+(0.06543564261548881*Exterior2nd_ImStu
cc)+(0.06289031153083985*Heating_GasA)+(0.04965531502704038*MasVnrType_Sto
ne)+(0.0491222551924809*LotConfig_CulDSac)+(0.047365218853938795*LandSlope
_Mod)+(0.04619250796887325*MSZoning_RL)+(0.044006000406870756*Fence_MnPrv)
+(0.04076468624124307*GarageCars)+(0.03636075916105678*MSSubClass_70)+(0.0
3449638539019934*MasVnrArea)+(0.033365939084722344*LotArea)+(0.03322100674
3903374*TotRmsAbvGrd)+(0.03150474325486273*BsmtCond_TA)+(0.030995025531134
677*PoolArea)+(0.030860587146145434*Neighborhood_SawyerW)+(0.0308054691037
11472*Foundation_PConc)+(0.029842663107472796*ScreenPorch)+(0.029592632993
17955*FullBath)+(0.0276031420964706*GarageArea)+(0.02530151677959752*LotFr
ontage)+(0.024539532456982156*Fireplaces)+(0.024400860059433985*LotShape_I
R2)+(0.021372577115344314*LandContour_HLS)+(0.019513274267286405*BsmtFinTy
pe1_Unf)+(0.017344152368795305*2ndFlrSF)+(0.017133460098988727*Condition2_
Norm)+(0.01678292450204316*BsmtFinSF2)+(0.01653001819448672*ExterCond_TA)+
(0.015793476202382695*CentralAir_Y)+(0.014738849460985737*HalfBath)+(0.014
433037007861811*MSSubClass_30)+(0.014119519105257048*WoodDeckSF)+(0.013938
920104395644*Fence_No
Fence)+(0.013903515728203173*OpenPorchSF)+(0.012474338631908773*HouseStyle
_1Story)+(0.012400540080163926*BsmtFullBath)+(0.010607628538533982*Exterio
r2nd_VinylSd)+(0.0077684370813286665*Exterior2nd_Wd
Sdng)+(0.007734446165380973*Condition1_Feedr)+(0.006601732776453101*3SsnPo
rch)+(0.006005366251749092*Street_Pave)+(0.005004644875210068*Neighborhood
_BrDale)+(0.004504799690746425*EnclosedPorch)+(0.004033364737209701*LotSha
```

pe_Reg)+(0.0037134029069846334*LotConfig_Inside)+(0.00287376649954151*GarageType_BuiltIn)+(0.001072220707685714*GarageType_No Garage)+(0.0010094547177121737*Neighborhood_NPkVill)+(6.743248620244689e-05*GarageCond_No Garage)+(2.465615337078277e-16*GarageFinish_No Garage)+(2.187538419362682e-16*GarageQual_No Garage)+(-0.0*BsmtUnfSF)+(0.0*1stFlrSF)+(0.0*MSSubClass_40)+(0.0*MSSubClass_45)+(0.0*MSSubClass_50)+(0.0*MSSubClass_60)+(-0.0*MSSubClass_80)+(-0.0*MSSubClass_85)+(-0.0*MSSubClass_90)+(-0.0*MSSubClass_160)+(-0.0*MSSubClass_180)+(-0.0*MSSubClass_190)+(0.0*MSZoning_RH)+(-0.0*MSZoning_RM)+(-0.0*Alley_No Alley)+(0.0*Alley_Pave)+(0.0*LotShape_IR3)+(0.0*LandContour_Lvl)+(-0.0*Utilities_NoSeWa)+(-0.0*LotConfig_FR2)+(-0.0*LotConfig_FR3)+(-0.0*Neighborhood_ClearCr)+(-0.0*Neighborhood_CollgCr)+(-0.0*Neighborhood_Gilbert)+(-0.0*Neighborhood_IDOTRR)+(0.0*Neighborhood_SWISU)+(0.0*Neighborhood_Somerst)+(-0.0*Neighborhood_Timber)+(0.0*Neighborhood_Veenker)+(0.0*Condition1_PosA)+(-0.0*Condition1_PosN)+(0.0*Condition1_RRAn)+(-0.0*Condition1_RRNe)+(-0.0*Condition1_RRNn)+(0.0*Condition2_Feedr)+(-0.0*Condition2_PosA)+(-0.0*Condition2_RRAe)+(-0.0*Condition2_RRAn)+(-0.0*BldgType_2fmCon)+(-0.0*BldgType_Duplex)+(0.0*HouseStyle_2.5Fin)+(-0.0*HouseStyle_2.5Unf)+(-0.0*HouseStyle_SFoyer)+(-0.0*HouseStyle_SLvl)+(0.0*OverallQual_2)+(-0.0*OverallQual_6)+(-0.0*OverallCond_2)+(0.0*OverallCond_6)+(0.0*RoofStyle_Gambrel)+(0.0*RoofStyle_Hip)+(-0.0*RoofStyle_Mansard)+(0.0*RoofStyle_Shed)+(-0.0*RoofMatl_Roll)+(0.0*RoofMatl_WdShake)+(0.0*RoofMatl_WdShngl)+(-0.0*Exterior1st_BrkComm)+(-0.0*Exterior1st_CBlock)+(-0.0*Exterior1st_CemntBd)+(-0.0*Exterior1st_ImStucc)+(0.0*Exterior1st_MetalSd)+(-0.0*Exterior1st_Stone)+(0.0*Exterior1st_Stucco)+(0.0*Exterior1st_VinylSd)+(0.0*Exterior1st_Wd Sdng)+(0.0*Exterior1st_WdShing)+(0.0*Exterior2nd_AsphShn)+(0.0*Exterior2nd_Brk Cmn)+(-0.0*Exterior2nd_BrkFace)+(-0.0*Exterior2nd_CBlock)+(-0.0*Exterior2nd_HdBoard)+(0.0*Exterior2nd_MetalSd)+(-0.0*Exterior2nd_Other)+(0.0*Exterior2nd_Stone)+(0.0*Exterior2nd_Stucco)+(0.0*MasVnrType_None)+(0.0*ExterCond_Fa)+(-0.0*ExterCond_Po)+(0.0*Foundation_Slab)+(0.0*Foundation_Stone)+(-0.0*Foundation_Wood)+(-0.0*BsmtQual_Fa)+(0.0*BsmtQual_No Basement)+(0.0*BsmtCond_Gd)+(0.0*BsmtCond_No Basement)+(-0.0*BsmtCond_Po)+(0.0*BsmtExposure_No Basement)+(-0.0*BsmtFinType1_BLQ)+(0.0*BsmtFinType1_No Basement)+(-0.0*BsmtFinType2_BLQ)+(0.0*BsmtFinType2_GLQ)+(0.0*BsmtFinType2_No Basement)+(-0.0*BsmtFinType2_Rec)+(0.0*BsmtFinType2_Unf)+(-0.0*Heating_GasW)+(-0.0*Heating_Grav)+(-0.0*Heating_OthW)+(0.0*Heating_Wall)+(0.0*HeatingQC_Fa)+(0.0*Electrical_FuseF)+(0.0*Electrical_FuseP)+(-0.0*Electrical_Mix)+(0.0*Electrical_SBrkr)+(-0.0*Functional_Maj2)+(0.0*Functional_Min1)+(0.0*Functional_Min2)+(-0.0*Functional_Mod)+(-0.0*Functional_Sev)+(-0.0*FireplaceQu_Fa)+(0.0*FireplaceQu_Gd)+(0.0*FireplaceQu_No

Fireplace)+(0.0*FireplaceQu_Po)+(-0.0*FireplaceQu_TA)+(-
0.0*GarageType_Attchd)+(0.0*GarageType_Detchd)+(0.0*GarageFinish_RFn)+(0.0
*GarageFinish_Unf)+(-0.0*GarageQual_Fa)+(-0.0*GarageQual_Gd)+(-
0.0*GarageQual_Po)+(-0.0*GarageQual_TA)+(-0.0*GarageCond_Gd)+(-
0.0*GarageCond_Po)+(0.0*GarageCond_TA)+(-0.0*PavedDrive_P)+(-
0.0*PavedDrive_Y)+(-0.0*PoolQC_Fa)+(-0.0*PoolQC_Gd)+(0.0*PoolQC_No
Pool)+(0.0*Fence_GdWo)+(-0.0*SaleType_CWD)+(0.0*SaleType_Con)+(-
0.0*SaleType_ConLD)+(-
0.0*SaleType_ConLI)+(0.0*SaleType_ConLw)+(0.0*SaleType_Oth)+(-
0.0*SaleType_WD)+(0.0*SaleCondition_AdjLand)+(0.0*SaleCondition_Alloca)+(-
0.0*SaleCondition_Family)+(0.0*SaleCondition_Partial)+(-
6.161777886553055e-05*GarageType_Basment)+(-
0.0003918036009840144*MiscVal)+(-0.00043523164600795314*OverallQual_4)+(-
0.000786507323712526*BsmtFinType1_Rec)+(-
0.003682864318188218*Neighborhood_Sawyer)+(-
0.004682983162225825*Neighborhood_OldTown)+(-
0.005848464971530954*MoSold)+(-0.006419282485575719*OverallQual_5)+(-
0.007048320220238*Neighborhood_NAmes)+(-
0.008690113027406942*Exterior1st_Plywood)+(-
0.009071732895797 49*BsmtHalfBath)+(-0.009419049694761978*Fence_MnWw)+(-
0.011457149772390847*Exterior2nd_CmentBd)+(-
0.014998415874478192*ExterCond_Gd)+(-
0.016799898871519354*OverallQual_3)+(-
0.01734135774501065*HouseStyle_2Story)+(-
0.019561728463139937*Neighborhood_MeadowV)+(-
0.020630129933506 1*RoofStyle_Gable)+(-
0.022169284260064304*BsmtExposure_Mn)+(-
0.02230964320915202*MSSubClass_120)+(-
0.023057747224715288*Exterior1st_HdBoard)+(-
0.026157430940471054*HeatingQC_TA)+(-
0.026480136916352346*BsmtFinType2_LwQ)+(-
0.02738254985500499*Foundation_CBlock)+(-
0.02821256172130621*GarageType_CarPort)+(-
0.028514369572836543*LowQualFinSF)+(-
0.02919462675770679*Exterior2nd_Plywood)+(-
0.02973514494770045*age_remodel)+(-0.030642011276296354*MSSubClass_75)+(-
0.031004423071864365*BsmtFinType1_LwQ)+(-
0.031089868909327845*HeatingQC_Gd)+(-0.03213535361701555*ExterQual_Gd)+(-
0.03464325914393596*GarageCond_Fa)+(-0.042012519802557595*BedroomAbvGr)+(-
0.04580286816956109*MasVnrType_BrkFace)+(-
0.04920179200078721*Neighborhood_Edwards)+(-
0.050332256517485116*Neighborhood_NWAmes)+(-
0.0684133229605647*BsmtExposure_No)+(-0.06908464818227901*KitchenAbvGr)+(-
0.0730069282752927*ExterQual_Fa)+(-0.0753731697670907*Exterior2nd_Wd
Shng)+(-0.0800038725812979*LandContour_Low)+(-
0.08369813256136838*KitchenQual_Fa)+(-
0.0872291349958656 4*KitchenQual_TA)+(-0.09047006430235817*ExterQual_TA)+(-
0.0958002869788007 3*OverallCond_5)+(-0.10701102278914555*BsmtQual_TA)+(-
0.10790206518245324*BldgType_TwnhsE)+(-
0.1097901187682250 9*Condition1_RRAe)+(-

0.11061305696800447*Neighborhood_Mitchel)+(-0.11435714263170176*LandSlope_Sev)+(-0.1293832068541192*KitchenQual_Gd)+(-0.1319320540980895*BldgType_Twnhs)+(-0.13565271138247154*OverallCond_4)+(-0.1558893820504271 8*BsmtQual_Gd)+(-0.16920894523874266*age)+(-0.22360851521060024*OverallCond_3)+(-0.3434054197970656*RoofMatl_Tar&Grv)

Question 3

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

Answer:

Below are the Top5 Lasso model predictors.

'OverallQual_10' 'OverallQual_9' 'GrLivArea' 'Neighborhood_StoneBr' 'SaleType_New'

After dropping these and created new Model below are the top5 predictors.

```
[214] X_train_new = X_train.drop(['categorical__OverallQual_10', 'categorical__OverallQual_9' ,'numericaltransformer__GrLivArea' ,
                                 'categorical__Neighborhood_StoneBr', 'categorical__SaleType_New'],axis=1)
      X_test_new = X_test.drop(['categorical__OverallQual_10', 'categorical__OverallQual_9' ,'numericaltransformer__GrLivArea' ,
                                 'categorical__Neighborhood_StoneBr', 'categorical__SaleType_New'],axis=1)
```

```
lasso_new = Lasso(alpha=0.001)
lasso_new.fit(X_train_new,y_train)
print(getTop10SignificantFeatures(lasso_new.feature_names_in_,lasso_new.coef_)[0])
```

```
                  features  coefficient
124           OverallCond_9     0.343808
8                  2ndFlrSF     0.334168
77     Neighborhood_NoRidge     0.263747
7                  1stFlrSF     0.259503
222          Functional_Typ     0.228927
135     Exterior1st_BrkFace     0.224010
267   SaleCondition_Partial     0.218164
68     Neighborhood_Crawfor     0.207826
185         BsmtExposure_Gd     0.203705
78     Neighborhood_NridgHt     0.166739
```

OverallCond_9, 2ndFlrSF, Neighborhood_NoRidge, 1stFlrSF, Functional_Typ

Question 4

How can you make sure that a model is robust and generalisable? What are the implications of the same for the accuracy of the model and why?

Answer:

A model is robust when any variation in dataset doesn't affects it's performance much. A generalized model is able to adapt to new , previously unseen data. To make model robust and generalized model should not be overfit. To make sure model is not overfitted we need to add penalty . A overfit model will have low bias (high accuracy) but high variance on unseen data. A robust and generalized model should not be too complex and overfit. To have balance between model complexity and accuracy , we have Ridge and Lasso regularization which shrinks the coefficients near to zero by Ridge and Lasso makes some co-efficient to zero.