ASSIGNMENT

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

   Optimum value of alpha for Ridge is 10
   Optimum value of alpha for Lasso is 0.001
   After doubling the value of alpha for Ridge and Lasso
   Ridge: alpha = 20
   R2 value for train decreases from 0.93 to 0.92 while R2 for test remains same
   Lasso: alpha = 0.002
   R2 value for train decreases from 0.92 to 0.90 while R2 for test remains same
   The Important predictors remains same with only difference with Foundation_PConc replaces Neighbourhood_Somerst but their coefficients decreases.
   Neighborhood_Crawfor, OverallQual, Neighborhood_BrkSide, OverallCond, Foundation_PConc

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?**

   In lasso, the penalty is the sum of the absolute values of the coefficients. Lasso shrinks the coefficient estimates towards zero and it has the effect of setting variables exactly equal to zero when lambda is large enough while ridge does not. Hence, much like the best subset selection method, lasso performs variable selection. The tuning parameter lambda is chosen by

cross validation. When lambda is small, the result is essentially the least squares estimates. As lambda increases, shrinkage occurs so that variables that are at zero can be thrown away. So, a major advantage of lasso is that it is a combination of both shrinkage and selection of variables. In cases with very large number of features, lasso allow us to efficiently find the sparse model that involve a small subset of the features.

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?**

   The five most important predictor variables after dropping earlier important predictors are:
   - Functional Type
   - KitchenQual Ex
   - MSZoning FV
   - LandContour_HLS
   - Exterior1st_BrkFace

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?**

   Robustness of model means the robust performance of the algorithm is the one which does not deteriorate too much when training and testing with slightly different data (either by adding noise or by taking another dataset), hence, algorithm is prone to overfitting. To improve Robustness following are the points:

1. **Testing Consistency with Specifications:** Techniques to test that machine learning systems are consistent with properties (such as invariance or robustness) desired by the designer and users of the system.

2. **Training Machine Learning models to be Specification-Consistent:** Even with copious training data, standard machine learning algorithms can produce predictive models that make predictions inconsistent with desirable specifications like robustness or fairness — this requires us to reconsider training algorithms that produce models that not only fit training data well, but also are consistent with a list of specifications.

3. **Formally Proving that Machine Learning Models are Specification-Consistent:** There is a need for algorithms that can verify that the model predictions are provably consistent with a specification of interest for all possible inputs. While the field of formal verification has studied such algorithms for several decades, these approaches do not easily scale to modern deep learning systems despite impressive progress.

   To improve Accuracy,

Hyperparameter Tuning

Hyperparameters in Machine Learning are user-controlled “settings” of your ML model. They influence how your model’s parameters will be updated and learned during training. Of course, the output of your model depends on its learned parameters, and its learned parameters are constantly updated and determined during the training phase. That updating is controlled by the model’s training, which is in turn influenced by the hyperparameters. Thus, if you can set the right hyperparameters, your model will learn the most optimal weights that it possibly can with a given training algorithm and data.

Finding the best hyper-parameters is usually done manually. It’s a simple task of trial and error, with some intelligent guesstimating. You’ll simply try as many hyperparameter settings as you have time

for and see which one gives you the best results. You can narrow your search space just by having some rough idea of what good parameters might be. That's a matter of domain knowledge, insight into your data, and experience with ML.

Ensemble Methods

Ensemble is the ML technique of combining the predictions of multiple models at once. The idea is that the combined knowledge of these models will give a more accurate result than the knowledge of any single one of them.

To build an ensemble, simply train multiple different ML models on the same data for the same task. At inference time, you will apply all the models to your input individually. If your task is classification, you can combine the results using a simple per class voting scheme or take the prediction with the highest confidence. For regression, just average out the results.

Feature Engineering

Feature engineering involves the careful selection and possible manipulation of your data's features. The purpose of this is to feed your model only the most optimal form of input. If you can consistently give your model only the parts of the data it needs to make accurate predictions, then it doesn't have to deal with any extra noise that comes from the rest of the data.

If you apply Principal Component Analysis and find that one of your features have very low correlation with the output, then you probably don't need to be processing it. Some features are going to be intuitively not useful, such as the ID or perhaps recording date. Or maybe you only want certain features to be considered in the first place.

To give your model the best your data has to offer, do some data exploration to find out what information and features are needed for

predictions. Often, you'll find that your dataset comes with some extra features that are either redundant or don't contribute to the prediction at all.