2.3 Data Preprocessing Steps

Data preprocessing is the most important thing we observed in this task. In total we need to deal with three things: 1. Drop irrelevant or meaningless features 2. Fill none value with 0 for certain features. 3. Transform the datasets so they can be more friendly for training.

For the first step, we need to remove some variables based on the description mentioned in the campuswire platform. Moreover, we need to set “Mo\_Sold” and “Year\_Sold” as categorical variables based on their actual meaning. For data or features with extreme value, we run Winsorization smooth processing () so that they can become more compact.

For the second step, we fill 0 for the lost value in features “Garage\_Yr\_Blt” both for training datasets and testing datasets.

For the third step, we will run an assembly function PreProcessingMatrixOutput () to sort the variables name and transfer the given datasets into matrix.

2.6 Training of Model 2 ( Linear regression with Elasticnet penalty)

After we finished the data preprocessing for our training set, we build a linear regression model based on glmnet package in R, which accept the prediction features as input and sale price as output. After studying in-depth on course materials and spending plenty of time adjusting the parameters, we finally perform better than the threshold error. For the details of input parameters, they are listed here:

For cv. glmnet() function, we have:

1. train.x: features as predictors.
2. train.y: sale prices as response.
3. alpha=0.1: alpha is for the elastic net mixing parameter 𝛼, with range 𝛼∈[0,1]. We pick 𝛼=0.1.

After finishing the training steps, we can get our expected model for this training-testing pair. Therefore, we can use the min lambda for the prediction step.

For prediction() function, we have:

1. cv.out: it is our elasticnet penalty model.
2. s=cv.out$lambda.min: min lambda value of our model.
3. newx: the testing data we input.

After running the prediction function, we will have a table whose column name are “PID” and “Sale\_Prices”, which mean we finished our training-prediction process.