MACHINE LEARNING

In Q1 to Q11, only one option is correct, choose the correct option:

1. Which of the following methods do we use to find the best fit line for data in Linear Regression?

ANS: - A) Least Square Error

2. Which of the following statement is true about outliers in linear regression?

ANS: - A) Linear regression is sensitive to outliers

3. A line falls from left to right if a slope is ______?

ANS: - B) Negative

4. Which of the following will have symmetric relation between dependent variable and independent variable?

ANS: - C) Both of them

5. Which of the following is the reason for over fitting condition?

ANS: - C) Low bias and high variance

6. If output involves label, then that model is called as:

ANS: - B) Predictive modal

7. Lasso and Ridge regression techniques belong to _________?

ANS: - D) Regularization

8. To overcome with imbalance dataset which technique can be used?

ANS: - D) SMOTE

9. The AUC Receiver Operator Characteristic (AUCROC) curve is an evaluation metric for binary classification problems. It uses _____ to make graph?

ANS: - C) Sensitivity and Specificity

10. In AUC Receiver Operator Characteristic (AUCROC) curve for the better model area under the curve should be less.

ANS: - B) False

11. Pick the feature extraction from below:

ANS: - D) Forward selection

In Q12, more than one options are correct, choose all the correct options:

12. Which of the following is true about Normal Equation used to compute the coefficient of the Linear Regression?

ANS: - A) We don’t have to choose the learning rate.

B) It becomes slow when number of features is very large.

Q13 to Q15 are subjective answer type questions, Answer them briefly.

13. Explain the term regularization?

ANS: - The method of regularization is very popular in the field of machine learning however you will see that many people are still not using it. One reason I can think of is because of the complexity behind the whole concept of the regularization so I thought to make it simple for all of us. In this article I am going to try to explain the regularization in a way that it is easy to understand and easy to use. Basically, while I explain the concept, I will give practical details t on how to implement regularization in R and SAS.

So, what is Regularization? In very simple terms Regularization refers to the method of preventing overfitting, by explicitly controlling the model complexity. It leads to smoothening of the regression line and thus prevents overfitting. It does so by penalizing the bent of the regression line that tries to closely match the noisy data points.

I know that if you don’t know what is overfitting then it can be confusing to you but if I start explaining what is overfitting and why it happens then this article will be about that only. I will write another post on overfitting but here in this past let us focus on Regularization only.

In Short Overfitting occurs when there are so many free parameters that the learning algorithm can fit the training data too closely which increase the generalization error and Regularization solve this problem of overfitting. There are couple of techniques to achieve regularization i.e., L1 and L2. Basically, you penalize your loss function by adding a multiple of an L1 (LASSO) or an L2 (Ridge) norm of your weights vector w (it is the vector of the learned parameters in your regression). Though both the Lasso(L1) and Ridge(L2) is a regularization technique which penalize the coefficients, but both have different properties and get used in different use cases. If you’ve used these techniques before then you must be knowing that how they penalizing the magnitude of coefficients of features along with minimizing the error between predicted and actual observations. So, the key difference is in how they assign penalty to the coefficients: -

$$\text{L1:} \quad R(\theta)=\|\theta\|_1=\sum_{i=1}^{n}|\theta_i|$$

$$\text{L2:} \quad R(\theta)=\|\theta\|_2^2=\sum_{i=1}^{n}\theta_i^2$$

R(theta) is the regularization term, which forces the parameters to be small.

In Lasso(L1) as you can see in the above formula that it adds penalty equivalent to absolute value of the magnitude of coefficients whereas in Ridge(L2) it adds penalty equivalent to square of the magnitude of coefficients and this is the basic different between both. Now the question will come in our mind that why do we need to penalize the coefficients. May be to answer it will be better if revisit any of your multi polynomial regression model and you will realize that as you can increase the complexity of the model the size of the coefficients also gets increased. So, what does large coefficients signifies? Basically, it means that we’re putting a lot of emphasis on that feature means the particular feature is a good predictor for the outcome. When it becomes too large the algorithm starts modelling complex relations to estimate the output and ends up overfitting to the particular training data and as I explained above that to solve the overfitting problem, we need regularization so I hope now it is clear that why we need to penalize the coefficients. Please feel free to reach in case it is still not clear and I am happy to discuss the same.

Now let us go in details on each regularization technique: -

RIDGE(L2): -As mentioned that it adds a factor of sum of squares of coefficients in the optimization objective. So, ridge regression optimizes the following f(x) = RSS + α * (sum of square of coefficients) Here, α (alpha) is the parameter which balances the amount of emphasis given to minimizing RSS vs minimizing sum of square of coefficients. α can take various values:

α = 0: The objective becomes same as simple linear regression. We’ll get the same coefficients as simple linear regression.

α = ∞: The coefficients will be zero because of infinite weightage on square of coefficients, anything less than zero will make the objective infinite.

0 < α < ∞: The magnitude of α will decide the weightage given to different parts of objective.

Ridge includes all of the features in the model. Thus, the major advantage of ridge regression is coefficient shrinkage and reducing model complexity.It is majorly used to prevent overfitting.

Since it includes all the features, it is not very useful in case of large feature. It generally works well even in presence of highly correlated features as it will include all of them in the model but the coefficients will be distributed among them depending on the correlation

LASSO(L1): -LASSO stands for Least Absolute Shrinkage and Selection Operator. So, lasso regression optimizes the following: f(x) = RSS + α * (sum of absolute value of coefficients) Here, α (alpha) works similar to that of ridge. When we will use L1 then we will see that even for a small value of alpha significant number of coefficients are zero. This fundament of most of the coefficients being zero is called ‘sparsity 'and that’s why lasso performs feature selection. So along with shrinking coefficients, lasso performs feature selection as well. As you can see that as per the formula some of the coefficients will become zero which means that particular feature gets excluded from Model. Lasso is preferred when we very large number of features.

When there is correlated variables, Lasso selects any one feature among the highly correlated ones and reduced the coefficients of the rest to zero. Also, the chosen variable changes randomly with change in model parameters. This generally doesn’t work that well as compared to ridge regression.

So the L2 penalty cause pure shrinkage whereas L1 penalty does shrinkage and selection.

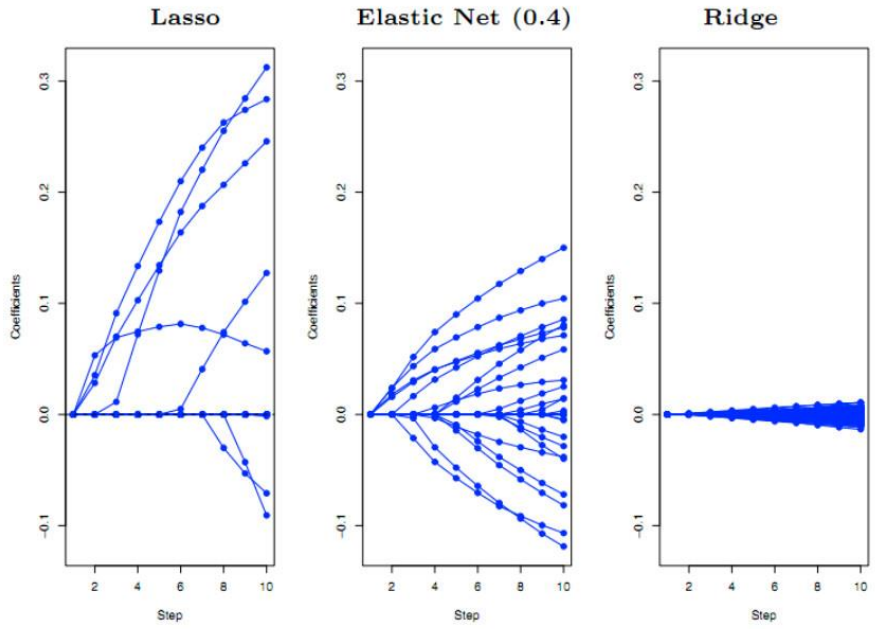
Along with Ridge and Lasso, Elastic Net is another useful technique which combines both L1 and L2 regularization. So Elastic Net will be used to balance the things from L1 & L2 which is explained above. Basically, Ridge regression, LASSO, and elastic net are part of the same family with penalty term:

$$P_\alpha = \sum_{i=1}^{p} \left[\frac{1}{2}(1-\alpha) b_j^2 + \alpha |b_j| \right]$$

Where:-

- α= 0 = ridge regression - Ridge regression shrinks correlated variables toward each other.
- α= 1 = LASSO- LASSO also does feature selection: – if many features are correlated, lasso will just pick one.
- 0 < α < 1 = elastic net!- Elastic net can deal with grouped variables.

Let us visualize it:-



Regularization in R:-

GLMNET is the mostly used R package for implementing regularization in R and my favorite one too.

install.packages("glmnet")

library(glmnet)

The main function in this package is glmnet(), which can be used to t ridge regression models, lasso models, and more. This function has slightly different syntax from other model fitting functions. The glmnet() function has an alpha argument that determines what type of model is fit. If alpha = 0 then a ridge regression model is fit, and if alpha = 1 then a lasso model is fit. Let us first fit a Ridge regression model:-

ridge.model=glmnet(x,y,alpha=0,lambda=grid)

Note that by default, the glmnet() function standardizes the variables so that they are on the same scale. To turn off this default setting, use the argument standardize = FALSE.This is how you can see the coefficients:-dim(coef(ridge.model))

We expect the coefficient estimates to be much smaller, in terms of l2 norm, when a large value of Lamda is used, as compared to when a small value of Lamda is used.

In order to fit a lasso model, we again use the glmnet() function; however,this time we use the argument alpha = 1.

cv.out=cv.glmnet(x[train,],y[train],alpha=1) # Fit lasso model on training data

```
plot(cv.out) # Draw plot of training MSE as a function of lambda

bestlam=cv.out$lambda.min # Select lamda that minimizes training MSE

lasso.pred=predict(lasso.mod,s=bestlam,newx=x[test,]) # Use best lambda to predict test data

lasso.coef[lasso.coef!=0] # Display only non-zero coefficient
```

How I use it, I use glmnet function which doesk -fold cross-validation for glmnet, produces a plot, and returns a value for lambda, first I use alpha=1 which means Lasso to get the significant variables:-

```
# Finding significant variables

feature_lasso <- cv.glmnet(x,y family = "binomial",

                     type.measure = "auc", parallel = TRUE,

                     nfolds = nfolds, maxit = 10000, alpha = 1)
```

How does it work as per R documentation: -The function runs glmnet nfolds+1 times; the first to get the lambda sequence, and then the remainder to compute the fit with each of the folds omitted. The error is accumulated, and the average error and standard deviation over the folds is computed. Note that cv.glmnet does NOT search for values for alpha. A specific value should be supplied, else alpha=1 is assumed by default. If users would like to cross-validate alpha as well, they should call cv.glmnet with a pre-computed vector foldid, and then use this same fold vector in separate calls to cv.glmnet with different values of alpha. Note also that the results of cv.glmnet are random, since the folds are selected at random. Users can reduce this randomness by running cv.glmnet many times, and averaging the error curves.

```
feature <- predict(feature_lasso, s = "lambda.min", type = "coefficients")
```

Then I re run the model using Ridge regression but only with significant varaibles which I got from above step:-

```
  cv.glmnet(x,y,

      family = "binomial", type.measure = "auc",

      parallel = run_parallel, foldid = foldid, maxit = 10000,

      alpha = a))
```

As explained above while running this I have pre computed the foldbid and then use the same in in separate calls to cv.glmnet with different values of alpha. Once I get the minimum alpha then using that best alpha (i.e. minimum for all iterations) I fit the model using elastic net:-

```
min_alpha_mod <- elastic net[[which.min(results$mins)]]
```

That’s how I get the model where with cross validation I applied Regularization.

Regularization in SAS: -In SAS we have PROC GLMSELECT which I used to implement Regularization:-

```
proc glmselect data=train plots=all;
partition fraction(validate=.3);
class c1 c2;
model y = c1|c2|x1|x2|x3|x4|x5 @2
/ selection=lasso (stop=none choose=validate);
run;
```

Where the MODEL statement request that a linear model be built using all the effects (c1, c2, x1, x2, x3, x4 and x5) and their two-way interactions. The PARTITION statement randomly reserves 30% of the data as validation data and uses the remaining 70% as training data. The training set is used for fitting the models, and the validation set is used for estimating the prediction error for model selection. The SELECTION=LASSO option in the MODEL statement requests LASSO selection. The CHOOSE=VALIDATE sub option in the MODEL statement requests that validation data be used as the tuning method for the LASSO selection.

14. Which particular algorithms are used for regularization?

ANS: - What is Regularization?

Regularization is a technique used in regression to reduce the complexity of the model and to shrink the coefficients of the independent features.

In simple words, this technique converts a complex model into a simpler one, so as to avoid the risk of overfitting and shrinks the coefficients, for lesser computational cost.

What are the different Regularization algorithms?

- Ridge Regression
- LASSO (Least Absolute Shrinkage and Selection Operator) Regression
- Elastic-Net Regression

Working of Ridge, LASSO, and Elastic-Net Regression

The working of all these algorithms is quite similar to that of Linear Regression, it's just the loss function that keeps on changing!

$$Loss = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} (y_i - (w_i x_i + c))^2$$

Loss Function for Linear Regression

Ridge Regression

Ridge regression is a method for analyzing data that suffer from multi-collinearity.

$$Loss = \sum_{i=1}^{n} (y_i - (w_i x_i + c))^2 + \lambda \sum_{i=1}^{n} w_i^2$$

Loss Function for Ridge Regression

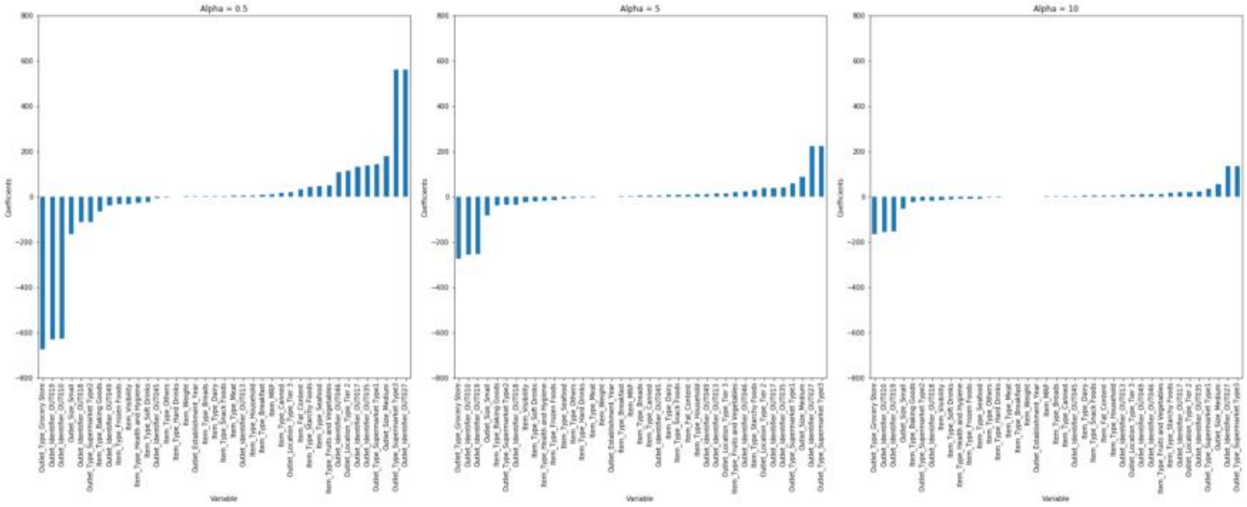
Ridge regression adds a penalty (L2 penalty) to the loss function that is equivalent to the square of the magnitude of the coefficients.

The regularization parameter (λ) regularizes the coefficients such that if the coefficients take large values, the loss function is penalized.

- λ → 0, the penalty term has no effect, and the estimates produced by ridge regression will be equal to least-squares i.e. the loss function resembles the loss function of the Linear Regression algorithm. Hence, a lower value of λ will resemble a model close to the Linear regression model.
- λ → ∞, the impact of the shrinkage penalty grows, and the ridge regression coefficient estimates will approach zero (coefficients are close to zero, but not zero).

Note: Ridge regression is also known as the L2 Regularization.

To sum up, Ridge regression shrinks the coefficients as it helps to reduce the model complexity and multi-collinearity.



Ridge Regression: Coefficient values if λ = 0.5, 5 and 10 respectively || Source

LASSO Regression

LASSO is a regression analysis method that performs both feature selection and regularization in order to enhance the prediction accuracy of the model.

$$Loss = \sum_{i=1}^{n} (y_i - (w_i x_i + c))^2 + \lambda \sum_{i=1}^{n} |w_i|$$

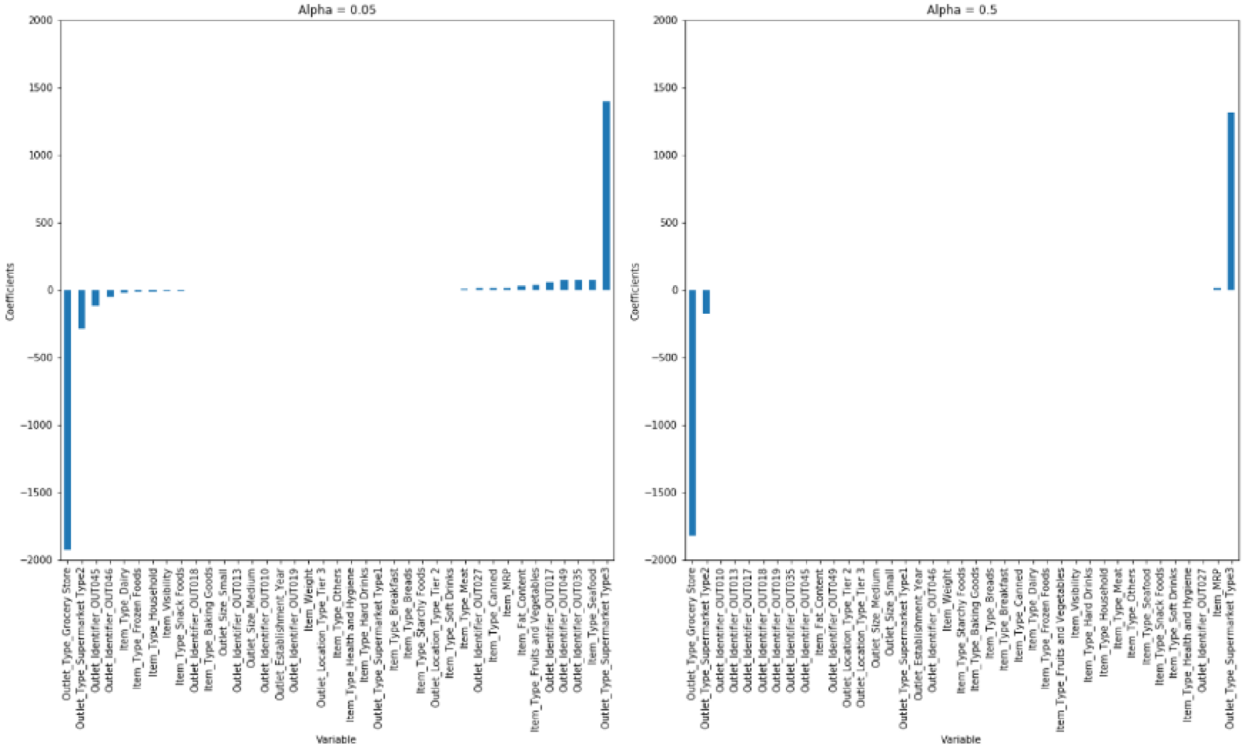
Loss Function for LASSO Regression

LASSO regression adds a penalty (L1 penalty) to the loss function that is equivalent to the magnitude of the coefficients.

In LASSO regression, the penalty has the effect of forcing some of the coefficient estimates to be exactly equal to zero when the regularization parameter λ is sufficiently large.

Note: LASSO regression is also known as the L1 Regularization (L1 penalty).

To sum up, LASSO regression converts coefficients of less important features to zero, which indeed helps in feature selection, and it shrinks the coefficients of remaining features to reduce the model complexity, hence avoiding overfitting.



Elastic-Net Regression

Elastic-Net is a regularized regression method that linearly combines the L1 and L2 penalties of the LASSO and Ridge methods respectively.

$$Loss = \sum_{i=0}^{n} (y_i - (w_i x_i + c))^2 + \lambda_1 \sum_{i=0}^{n} |w_i| + \lambda_2 \sum_{i=0}^{n} w_i^2$$

What does Regularization achieve?

A standard least-squares model tends to have some variance in it i.e., the model won't generalize well for a data set different than its training data. Regularization, significantly reduces the variance of the model, without a substantial increase in its bias.

So, the regularization parameter λ, used in the techniques described above, controls the impact on bias and variance. As the value of λ rises, it reduces the value of coefficients and thus reducing the variance. This increase in λ is beneficial as it is only reducing the variance (hence avoiding overfitting), without losing any important properties in the data. But after a certain value, the model starts losing important properties, giving rise to bias in the model and thus underfits the data. Therefore, the value of λ should be carefully selected.

This is all the basic you will need, to get started with Regularization. It is a useful technique that can help in improving the accuracy of your regression models.

15. Explain the term error present in linear regression equation?

ANS: - What Is an Error Term?

An error term is a residual variable produced by a statistical or mathematical model, which is created when the model does not fully represent the actual relationship between the independent variables and the dependent variables. As a result of this incomplete relationship, the error term is the amount at which the equation may differ during empirical analysis.

The error term is also known as the residual, disturbance, or remainder term, and is variously represented in models by the letters e, ε, or u.

KEY TAKEAWAYS

- An error term appears in a statistical model, like a regression model, to indicate the uncertainty in the model.

- The error term is a residual variable that accounts for a lack of perfect goodness of fit.
- Heteroskedastic refers to a condition in which the variance of the residual term, or error term, in a regression model varies widely.

Understanding an Error Term

An error term represents the margin of error within a statistical model; it refers to the sum of the deviations within the regression line, which provides an explanation for the difference between the theoretical value of the model and the actual observed results. The regression line is used as a point of analysis when attempting to determine the correlation between one independent variable and one dependent variable.

Error Term Use in a Formula

An error term essentially means that the model is not completely accurate and results in differing results during real-world applications. For example, assume there is a multiple linear regression function that takes the following form:

\begin{aligned} &Y = \alpha X + \beta \rho + \epsilon **Error! Hyperlink reference not valid.** &\textbf{where:} **Error! Hyperlink reference not valid.** &\alpha, \beta = \text{Constant parameters} **Error! Hyperlink reference not valid.** &X, \rho = \text{Independent variables} **Error! Hyperlink reference not valid.** &\epsilon = \text{Error term} **Error! Hyperlink reference not valid.** \end{aligned} Y=αX+βρ+ϵwhere: α, β=Constant parametersX,ρ=Independent variablesϵ=Error term When the actual Y differs from the expected or predicted Y in the model during an empirical test, then the error term does not equal 0, which means there are other factors that influence Y.

What Do Error Terms Tell Us?

Within a linear regression model tracking a stock’s price over time, the error term is the difference between the expected price at a particular time and the price that was actually observed. In instances where the price is exactly what was anticipated at a particular time, the price will fall on the trend line and the error term will be zero.

Points that do not fall directly on the trend line exhibit the fact that the dependent variable, in this case, the price, is influenced by more than just the independent variable, representing the passage of time. The error term stands for any influence being exerted on the price variable, such as changes in market sentiment.

The two data points with the greatest distance from the trend line should be an equal distance from the trend line, representing the largest margin of error.

If a model is heteroskedastic, a common problem in interpreting statistical models correctly, it refers to a condition in which the variance of the error term in a regression model varies widely.

Linear Regression, Error Term, and Stock Analysis

Linear regression is a form of analysis that relates to current trends experienced by a particular security or index by providing a relationship between a dependent and independent variable, such as the price of a security and the passage of time, resulting in a trend line that can be used as a predictive model.

A linear regression exhibits less delay than that experienced with a moving average, as the line is fit to the data points instead of based on the averages within the data. This allows the line to change more quickly and dramatically than a line based on numerical averaging of the available data points.

The Difference Between Error Terms and Residuals

Although the error term and residual are often used synonymously, there is an important formal difference. An error term is generally unobservable and a residual is observable and calculable, making it much easier to quantify and visualize. In effect, while an error term represents the way observed data differs from the actual population, a residual represents the way observed data differs from sample population data.