Question 1

Rahul built a logistic regression model with a training accuracy of 97% and a test accuracy of 48%. What could be the reason for the gap between the test and train accuracies, and how can this problem be solved?

Answer:

The model has a train accuracy of 97% and a test accuracy of 48%. This means that it has memorized the training data but lost the generalizability and hence unable to perform well on unseen data. This is a situation of overfitting**.** Overfitting can be solved in the following ways:

1) Cross Validation → Cross-validation is a technique for evaluating models by training several models on subsets of the available input data and evaluating them on the complementary subset of the data. This can be achieved by the k-fold cross validation technique. In this, we partition the data into k subsets, called folds. Then, we iteratively train the algorithm on k-1 folds while using the remaining fold as the test set.
2) Regularization → Regularization strikes the balance between keeping the model simple and not making it too naïve to be of any use. The model can be simpler in following terms:-
    - Number of parameters required to specify the model completely.
    - The degree of function, if it is a polynomial. Lesser the degree, simpler is the model.
    - Size of the best-possible representation of model in bits.
    - The depth/size of a decision tree.

    For regression, regularization would mean adding a regularization term to the cost that adds up the absolute values or the squares of the parameters of the model
3) Train with more data- It won't work every time, but training with more data can help algorithms detect the signal better.

Question 2

List at least four differences in detail between L1 and L2 regularisation in regression.

Answer:

S.No	L1 Regression (Lasso)	L2 Regression (Ridge)
1	Lasso Regression adds "absolute value of magnitude" of coefficient as penalty term to the loss function. $\sum_{i=1}^{n}(Y_i - \sum_{j=1}^{p} X_{ij}\beta_j)^2 + \lambda\sum_{j=1}^{p}\lvert\beta_j\rvert$ Cost function	Ridge regression adds "squared magnitude" of coefficient as penalty term to the loss function. $\sum_{i=1}^{n}(y_i - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda\sum_{j=1}^{p}\beta_j^2$ Cost function
2	Lasso shrinks the less important feature's coefficient to zero thus, removing some feature altogether. Thereby helps in performing feature selection.	Ridge shrinks the coefficients to small values but not to exactly zero. Hence does not help in performing feature selection.

	Lasso has derivative as k (constant). The derivative of Lasso subtracts some constant from the coefficients every time and encourages the uninformative coefficients towards zero.	Ridge has derivative as 2β. The derivative of Ridge removes x% of the coefficient every time and doesn't normally drive coefficients towards zero.
3	Lasso Regression is computationally more intensive than Ridge. It requires iterations to get to the final solution.	It almost always has a matrix representation of the solution. That's why it has lower computational cost.
4	For same α, Lasso has higher RSS as compared to Ridge.	For same α, Ridge has lower RSS as compared to Lasso.
5	Lasso works best when the model contains a lot of useless variables.	Ridge works best when most of the variables in the model are useful.

Question 3

Consider two linear models:

L1: y = 39.76x + 32.648628

And

L2: y = 43.2x + 19.8

Given the fact that both the models perform equally well on the test data set, which one would you prefer and why?

Answer:

The model L2 will be preferred as it is simpler than L1 based on the size of the best-possible representation of the model. (See more explanation below)

According to Occam's razor, a predictive model has to be as simple as possible, but no simpler as the simple model will be more generalizable and robust.

The simplicity can be defined in following terms:-

- Number of parameters required to specify the model completely :
    - In a simple linear regression for the response attribute y on the explanatory attributes x1,x2,x3, the model y = ax1 +bx2 is 'simpler' than the model y = ax1 +bx2 +cx3. The latter requires 3 parameters compared to the 2 required for the first model.
- The degree of the function, if it is a polynomial:
    - Considering regression again, the model y = ax1^2 +bx2^3 would be a more complex model because it is a polynomial of degree 3.
- Size of the best-possible representation of the model (For instance the number of bits in a binary encoding of the model)

    - More complex (messy, too many bits of precision, large numbers, etc.) the coefficients in the model, more complex it is. For example the expression (0.552984567 ∗ x 2 + 932.4710001276) could be considered to be more 'complex' than say (2x +3x 2 +1), though the latter has more terms in it.
- The depth or size of a decision tree.

This, L2 can be simple in terms of binary encoding of the model as it will require less bits than the L1 model.

Question 4

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

Answer:

Per, Occam's Razor— given two models that show similar 'performance' in the finite training or test data, we should pick the one that makes fewer on the test data due to following reasons:-

- Simpler models are usually more 'generic' and are more widely applicable
- Simpler models require fewer training samples for effective training than the more complex ones and hence are easier to train.
- Simpler models are more robust.
    - Complex models tend to change wildly with changes in the training data set
    - Simple models have low variance, high bias and complex models have low bias, high variance
- Simpler models make more errors in the training set. Complex models lead to overfitting — they work very well for the training samples, fail miserably when applied to other test samples

Therefore to make the model more robust and generalizable, make the model simple but not simpler which will not be of any use.

Regularization can be used to make the model simpler. Regularization helps to strike the delicate balance between keeping the model simple and not making it too naive to be of any use. For regression, regularization involves adding a regularization term to the cost that adds up the absolute values or the squares of the parameters of the model.

Also, Making a model simple leads to Bias-Variance Trade-off:

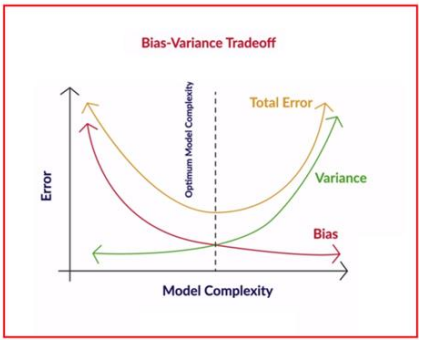
- A complex model will need to change for every little change in the dataset and hence is very unstable and extremely sensitive to any changes in the training data.
- A simpler model that abstracts out some pattern followed by the data points given is unlikely to change wildly even if more points are added or removed.

Bias quantifies how accurate is the model likely to be on test data. A complex model can do an accurate job prediction provided there is enough training data. Models that are too naïve, for e.g.,

one that gives same answer to all test inputs and makes no discrimination whatsoever has a very large bias as its expected error across all test inputs are very high.

Variance refers to the degree of changes in the model itself with respect to changes in the training data.

Thus accuracy of the model can be maintained by keeping the balance between Bias and Variance as it minimizes the total error as shown in the below graph



Question 5

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 optimal lambda value in case of Ridge and Lasso is as below:
  - Ridge - 10
  - Lasso - 0.0004
- The Mean Squared error in case of Ridge and Lasso are:
  - Ridge - 0.013743
  - Lasso - 0.013556
- The Mean Squared Error of Lasso is slightly lower than that of Ridge.
- Also, since Lasso helps in feature reduction (as the coefficient value of one of the feature became 0), Lasso has a better edge over Ridge.

Therefore, the variables predicted by Lasso can be applied to choose significant variables for predicting the price of a house.