Problem Statement - Part II

Question-1: Rahul built a logistic regression model having a training accuracy of 97% while the test accuracy was 48%. What could be the reason for the seeming gulf between test and train accuracy and how can this problem be solved.

Answer 1:

The reason for the seeming gulf between test and train accuracy is overfitting is a where a model performs well on training data but does not generalise well to test data. Overfitting happens when a model is hugely complex, such as having too many parameters relative to the number of observations. Such model has poor predictive performance, as it overreacts to minor changes in the training data. To avoid overfitting, it is necessary to use some techniques for example regularization & cross-validation. Regularization is a way of finding a good bias-variance tradeoff by tuning the complexity of the model. It is a very useful method to handle high correlation among features, filter out noise from data. Regularization is the simplification done by the training algorithm to control the model complexity. For example

1. For Regression it involves adding a regularization term to the cost which adds up the absolute values or the squares of the parameters of the model.

2. For decision trees this could mean 'pruning' the tree to control its depth and/or size.

3. For neural networks a common strategy is to include a drop out — dropping a few neurons and/or weights at random.

Regularization is a process used to create an optimally complex model, i.e. a model which is as simple as possible while performing well on the training data. Through regularization, the algorithm designer tries to strike the delicate balance between keeping the model simple, yet not making it too naive to be of any use. The regression does not account for model complexity - it only tries to minimize the error (e.g. MSE), although if it may result in arbitrarily complex coefficients. On the other hand, in regularized regression, the objective function has two parts - the error term and the regularization term.

Question-2: List at least 4 differences in detail between L1 and L2 regularization in regression.

Answer -2:

The main difference between L1 and L2 regularization is that L1 can yield sparse models while L2 cannot. Sparse model is a great property to have when dealing with high-dimensional data, for at least 2 reasons:

- Model compression

- Feature selection

2. Feature selection – L1 (Lasso) has in built feature selection whereas L2 (Ridge) doesn't do feature selection. Lasso trims down the coefficients of redundant variables to zero, and thus indirectly performs variable selection also. Ridge, on the other hand, reduces the coefficients to arbitrarily low values, though not zero.

3. In ridge regression, an additional term of "sum of the squares of the coefficients" is added to the cost function along with the error term. Whereas in In case of lasso regression, a regularisation term of "sum of the absolute value of the coefficients" is added

4. L1 (Lasso) regularization is computationally more intensive whereas ridge is computationally efficient. Ridge regression almost always has a matrix representation for the solution while Lasso requires iterations to get to the final solution.

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 dataset, which one would you prefer and why?

Answer 3:

I will select the L2: y = 43.2x + 19.8 model because of simplicity of the model.

More complex the model, less simple it is. Here are a few ways of looking the complexity of a model.

1. Number of parameters required to specify the model completely.

2. The degree of the function, if it is a polynomial.

Size of the best-possible representation of the model. For instance the number of bits in a binary encoding of the model. For instance more complex (messy, too many bits of precision, large numbers, etc.) the coefficients in the model, more complex it is. For example in the above example : 1. L1: y = 39.76x + 32.648628

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-4:

The simpler the model is, the more robust and generalisable it is. Occam's Razor is a simple thumb rule. It says given two models that show similar performance in the finite training or test data, we should pick the one that makes fewer assumptions about the data that is yet to be seen. That essentially means we need to pick the simpler of the two models.
1. Simpler models are usually more 'generic' and are more widely generalizable.

2. Simpler models require fewer training samples for effective training than the more complex ones and are consequently easier to train.
3. Simpler models are more robust - they are not as sensitive to the specifics of the training data set. So ideally the model must be immune to the specifics of the training data provided. 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. Here variance refers to the variance in the model and bias is the deviation from the expected ideal behaviour. It is referred to as the bias-variance tradeoff.
4. Simpler models make more errors in the training set, It is the price one pays for greater predictability. Complex models lead to overfitting — they work very well for the training samples, fail miserably when applied to other test samples.
5. We should use metrics which take into account both model fit and simplicity. They penalise the model for being too complex (i.e. for overfitting), and thus are more representative of the unseen 'test error'. Some examples of such metrics are Adjusted R 2, A I C and B I C
6. We can also estimate the test error via a validation set or a cross-validation approach.

Question-5: As you have determined the optimal value of lambda for ridge and lasso regression during the assignment, which one would you choose to apply and why?

Answer 5: We have already seen in the question 4 the difference between Lasso & Ridge. Based on that difference & the optimal value of lambda for ridge and lasso regression, I would choose lasso because:

1. Even though lasso is computationally more intensive, has in built feature selection. In the assignment we have not done RFE or any other feature elimination. . Lasso trimmed down the coefficients of redundant variables to zero, and thus indirectly performed variable selection.

2. Also if we will look at R square score of both the models, we can see that train & test R square score has less difference in the lasso model - Train (0.93) & Test (0.91).