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?
Ans:

This is typical case of overfitting the model where model used has learn most of the data from training data set or The size of the validation and /or the test set is too low

To solve this issue following can be approach .

1) Regularisation: Specifying an additional criterion that is traded of against fitting the training data. The Lasso (L1) or Ridge (L2) regularisation Often a pretty crude regularization which does a rather good job provided appropriated value for the regularization parameter is set and if we happen to have enough data the problem becomes even less critical.
2) Another way would be to define a regularization using hyperparameters and learn these to. This may be more robust in the case of parameter misspecification but effectively only shifts the problem to a higher level.
3) If this is not enough we can validate our learning procedure using techniques like cross-validation, which is a means to adjust the regularization. But this may be computationally expensive.
4) Increase the amount of sample in the training set.
5) Reduce the amount of feature.

Question 2
List at least four differences in detail between L1 and L2 regularisation in regression.

Answer:

"LASSO" L1-regularization	**"Ridge" L2-regularization**
The weights for each parameter are assigned as a 0 or 1 (binary value). This helps perform feature selection in sparse features spaces and is good for high-dimensional data since the 0 coefficient will cause some features to not be included in the final model	L2 regularization spreads the error among all the features. This results in weights for every feature with the possibility that some weights are really small values close to 0
L1 can also save on computational costs since the features weighted 0 can be ignored, however, model accuracy is often lost for this benefit	L2 tends to be more accurate in almost every situation however at a higher computational cost.
L1 is best used in high dimensional or sparse data sets when doing classification.	It is best used in non-sparse outputs, when no feature selection needs to be done, or if you need to predict a continuous output.

As an Error function L1 basically minimizing the sum of the absolute differences **(S)** between the target value (**Yi**) and the estimated values (**f(xi)**): $S = \sum_{i=1}^{n} \lvert y_i - f(x_i) \rvert.$	L2 basically minimizing the sum of the square of the differences **(S)** between the target value (**Yi**) and the estimated values (**f(xi):** $S = \sum_{i=1}^{n} (y_i - f(x_i))^2$

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?

Ans:

As mentioned the model perform equally on both test and training data so I would choose to L2 model due to more specific or rounded coefficient .

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?

Ans:

To create generalise model following step to be done.

1. **Check for linearity :**Carry out exploratory data analysis by examining scatter plots of explanatory and dependent variables, Choose an appropriate set of functions which seem to fit the plot well, build models using them, and compare the results.
2. **Feature Engineering**: Instead of using the raw explanatory variables in the current form, we create some function of the explanatory variables to best explain the data points. These functions capture the non-linearity in the data , The derived feature could be combination of two more attribute and/or **transformations of individual attributes**. These combinations and transformations could be **linear or non-linear.**
3. the basic algorithm remains the same as linear regression- we compute the values of constants which result in the least possible error (best fit). The only difference is that we now use the features instead of raw attribute.

The Implication of Generalised model on accuracy is it may lead to model complexity, overfitting/underfitting.

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:
Best alpha value for Lasso: {'alpha': 0.001}
Best alpha value for Ridge: {'alpha': 10}

After creating model in both Ridge and Lasso we can see that the r2_scores are almost same for both of them but as lasso will penalize more on the dataset and can also help in feature elimination (as seen in the Graph) I am going to consider that as my final model.