Advanced Regression Notes:

Summarising the important takeaways from the lecture:

1. In generalised regression models, the basic algorithm remains the same as linear regression. We compute the values of coefficients which result in the least possible error (best fit). The only difference is that we now use the features ϕ1(x),ϕ2(x),ϕ3(x)....ϕk(x) instead of the raw attributes.
2. The term 'linear' in linear regression refers to the linearity in the coefficients, i.e., the target variable y is linearly related to the model coefficients. It does not require that y should be linearly related to the raw attributes or features; feature functions could be non-linear as well.

‘Linear’ in Linear Regression:

 The model is called 'linear' because the targety**is linearly related to the** **coefficients**. To fully understand this, it is crucial to note that in regression, the**coefficients** a0,a1,a2,...,ak are your **variables**, i.e., you are trying to find the optimal coefficients that minimise some loss function. On the other hand, the **features** ϕ1(x),ϕ2(x),ϕ3(x)....ϕk(x) are actually **constants** because you are already given the dataset (i.e., the values of x, and hence ϕ(x), are fixed; so, what you are trying to tune are the coefficients).

Thus, saying that 'y is linearly related to the coefficients' implies that **only two operations**can be applied between the coefficients:

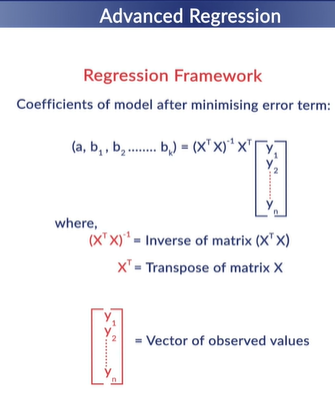
1) **Multiplying them by constants** (i.e., the features) such as a1ϕ1(x),a2ϕ2(x) and

2)**Adding the terms**with each other such asa1ϕ1(x)+a2ϕ2(x). What you **cannot** do is multiply them together, raise to one another's power, etc. That is, you cannot have terms such as a0.a1,aa32 etc.

Solving this system means to find the set of parameters a0,a1,...,a5 which satisfies all the equations. This can be done efficiently using **matrices**(and is done that way by many libraries)**.** The alternate way is to use optimisation methods such as gradient descent.

It turns out that the closest approximation of X−1 (for non-square matrices) is (XTX)−1XT. Thus, the (approximate) solution to this system is given by:





Summary:  
To summarise, you learnt that one needs to follow a 3-step process to build a regression model:

* First, we explored and visualized raw attributes to understand the shape of scatter plots.
* Second, we assessed which function of the explanatory variable would explain the shape of the data.
* Lastly, we wrote the generalized regression formula using the matrix format. We then summed up the errors between predicted and actual response variables and minimised the residual sum of error to arrive at the best-fit regression curve.

Another thing to note is that the term 'linear' in regression depicts the linear expression in the coefficients of the linear combination. It does not mean linear expression in raw attributes or features.

**Regularized regression**:

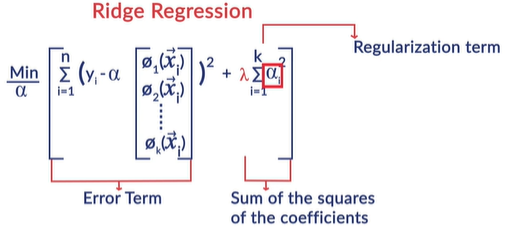
There are two kinds of regularization of a generic model, namely

* Ridge Regression
* Lasso Regression

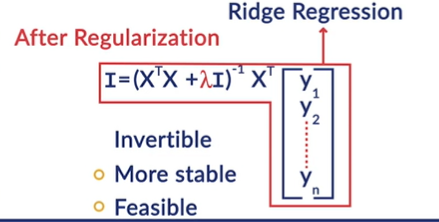
**Regularisation** is a process used to create an optimally complex model, i.e., a model that is as simple as possible while performing well on the training data. Through regularisation, one tries to strike the delicate balance between keeping the model simple and yet not making it too naive for any use.

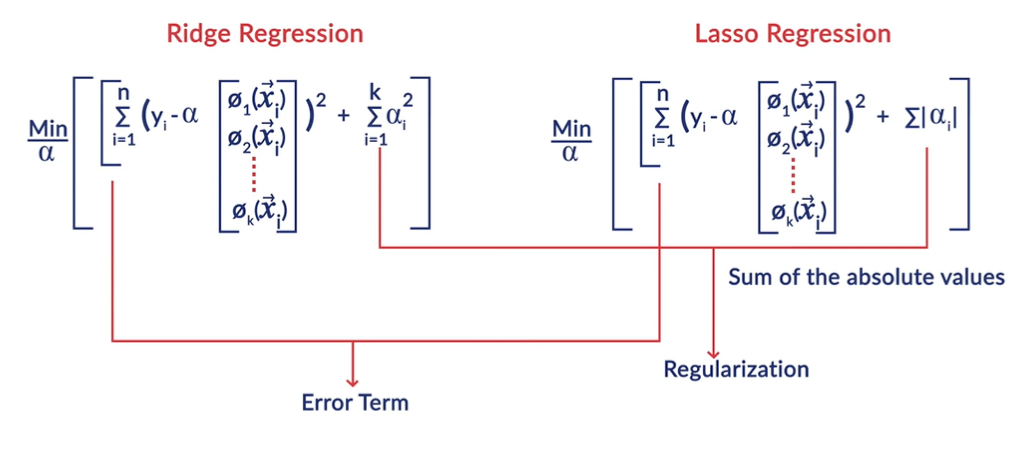
 Linear regression does not account for model complexity; it only tries to minimise the error (e.g., MSE). However, it may result in arbitrarily complex coefficients. On the other hand, in regularised regression, the objective function has two parts: the **error term** and the **regularisation term**.

1. Adding **sum of squares** of coefficients: *Ridge Regression*
2. Adding **Absolute value** of coefficients: *Lasso Regression*



Significance of Lambda in Ridge Regularized Regression:

* If the value of lambda used is high, any complexity will be clamped down
* If the value of lambda is closer to 0, then, unregularized regression is being performed (does not care about the complexity of the model)
* 



Lasso is more computationally intensive than Ridge Regression.

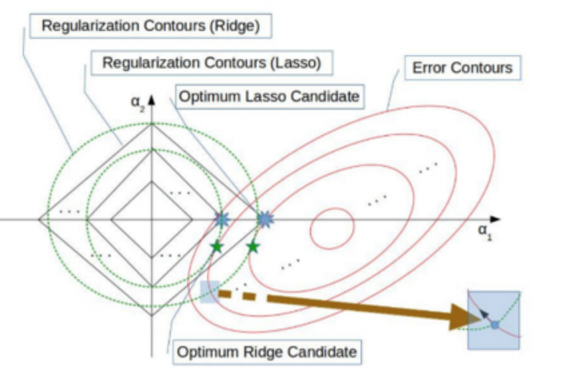
Both these methods are used to make a regression model simpler while balancing the 'bias-variance' trade-off.

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

Lasso Regression is helpful for Feature Selection.

To summarise, one of the most important benefits of **lasso regression**is that it results in model parameters, such that the lesser important features' coefficients, becoming zero. In other words, lasso regression indirectly performs **feature selection.**

Thus, the cost function for both ridge and lasso is given by:   

A **contour** of a function f(α) is a trace (locus) of the points that satisfy the equation f(α)=c for some constant c. The figure shows contours for various values of c: the 'inner' contours represent lower values of the error/regularisation terms. You want to find the coefficients that **minimise both the error term and the regularisation term.**

Thus, the key observation here is that at the optimum solution for α (the place where the sum of the error and regularisation terms is minimum), the corresponding regularisation contour and the error contour must 'touch' each other tangentially and not 'cross'.

*The outer contour represents the highest error value as described in the comprehension.*