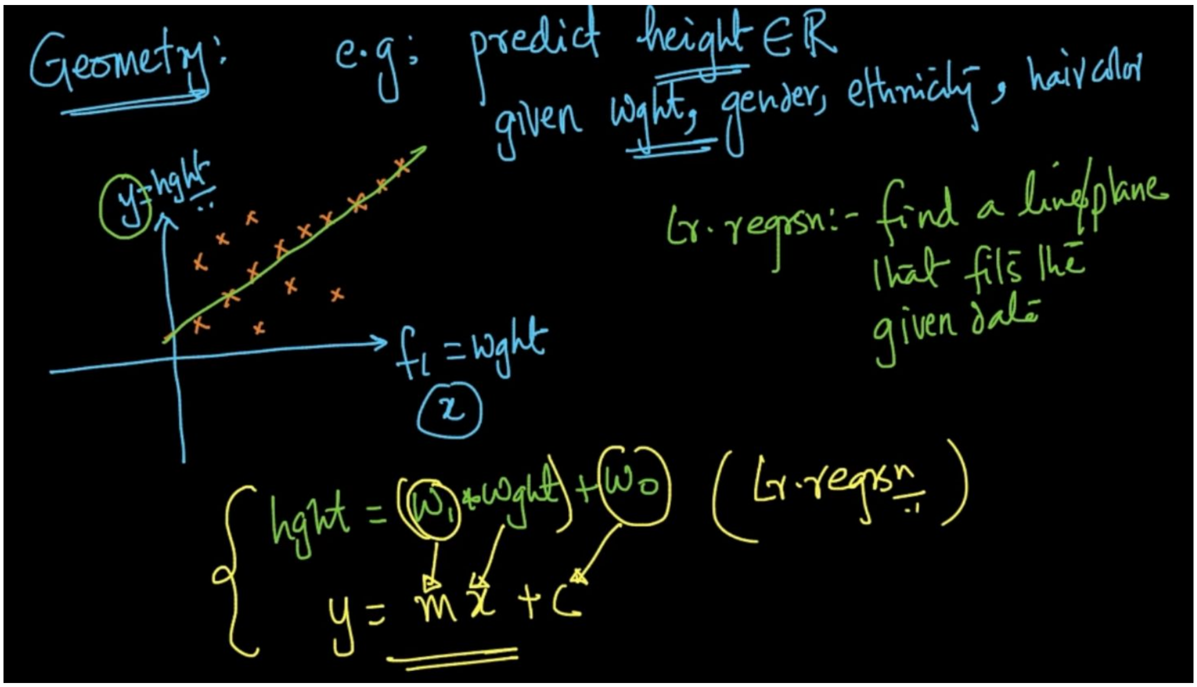
LINEAR REGRESSION

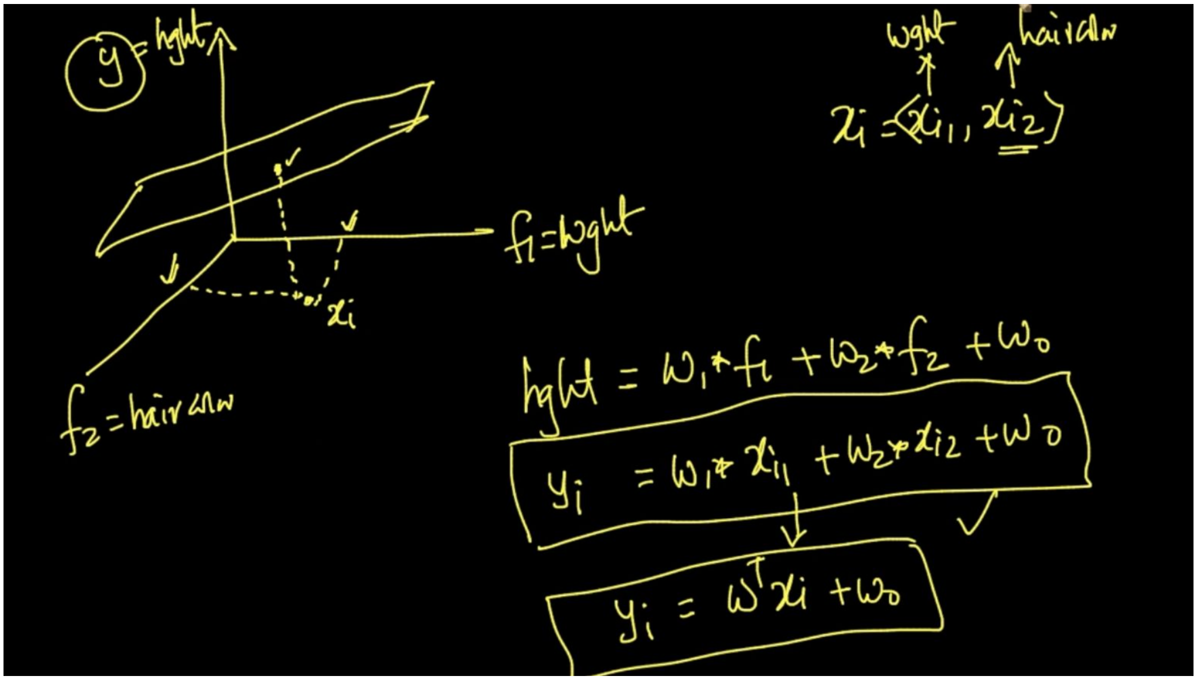
GEOMETRIC INTUITION BEHIND LINEAR REGRESSION

Linear Regression is one of the Regression Technique where y $\in R$ i.e Real number instead of a class label



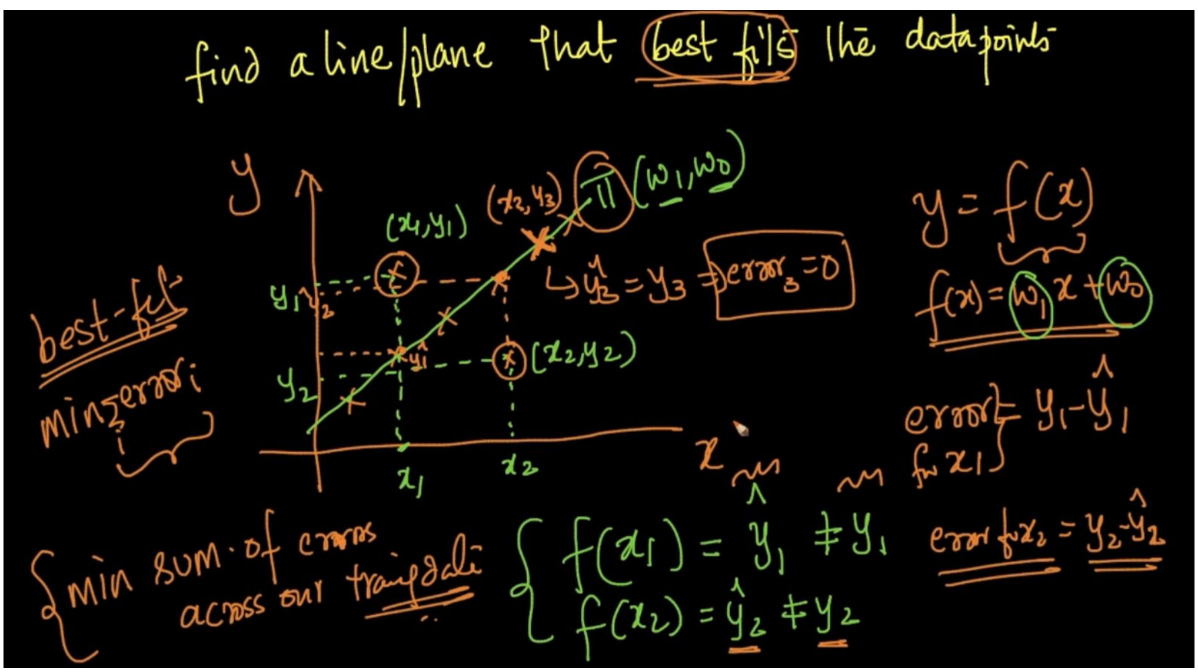
In Linear Regression we want the find the line which fits the given data. Ex: We want to predict height (y) given our features and we are using our feature weight (x).

Height = w_1 * weight + w_0 (Linear Regression). In this algorithm our goal is to find w_1, w_0 which can be compared to m(slope) and c(y-intercept) respectively



If we are predicting using two features. We'll plot the point using them and see their value in the plane .That'll be the value of y . In the above example, height = $w_1 * f_1 + w_2 * f_2$ + w_0

Therefore, if we want to predict y_i it'll be y_i = $w_1 * x_{i1} + w_2 * x_{i2}$ + w_0 => y_i = $w^T x_i$ + w_0



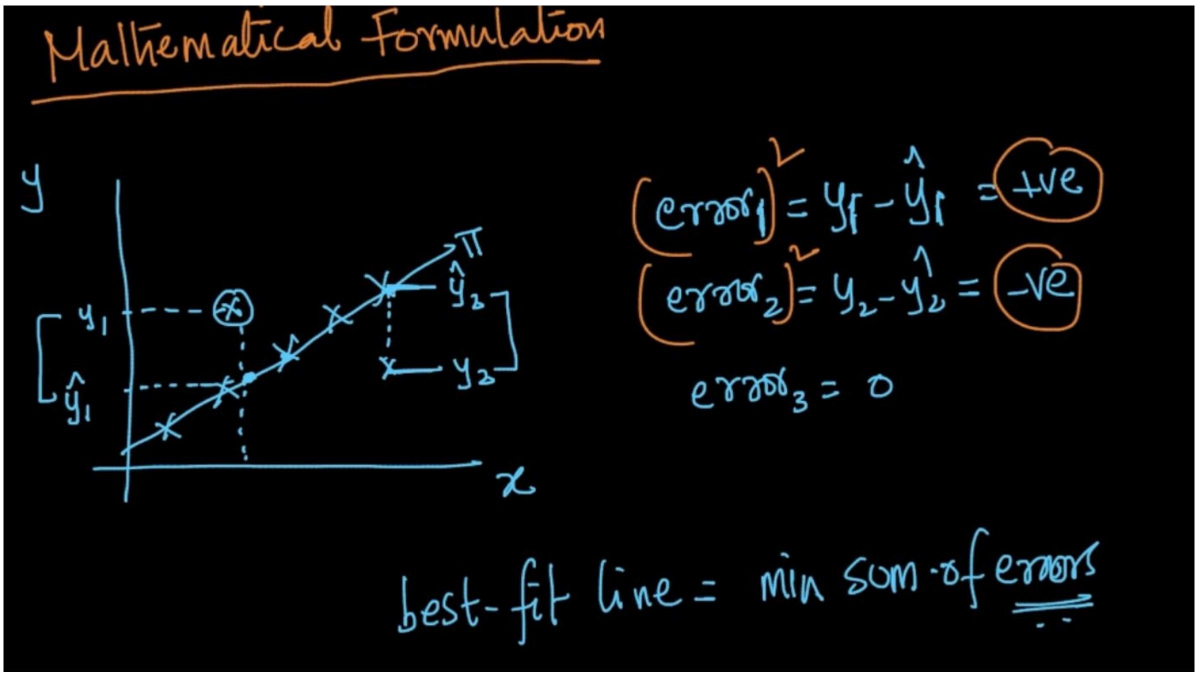
We've found the best fitting line to the data points i.e we've found w_1, w_0 .We've a point

(x_1, y_1) as we can see it's not lying in the line .If we apply our algorithm f(x_1) = $\widehat{y_1} \neq y_1$.

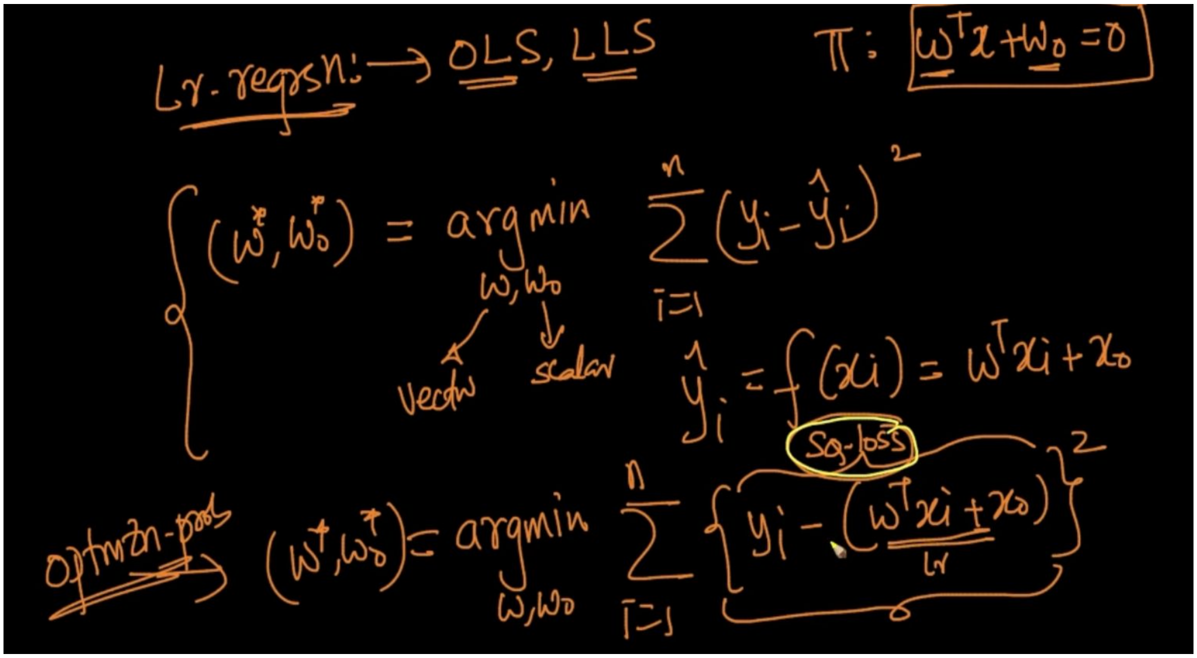
So, it's an error as predicted is not the same as actual label.Therefore, error = y_1 - $\widehat{y_1}$

We want to minimize the sum of errors across our Training Data

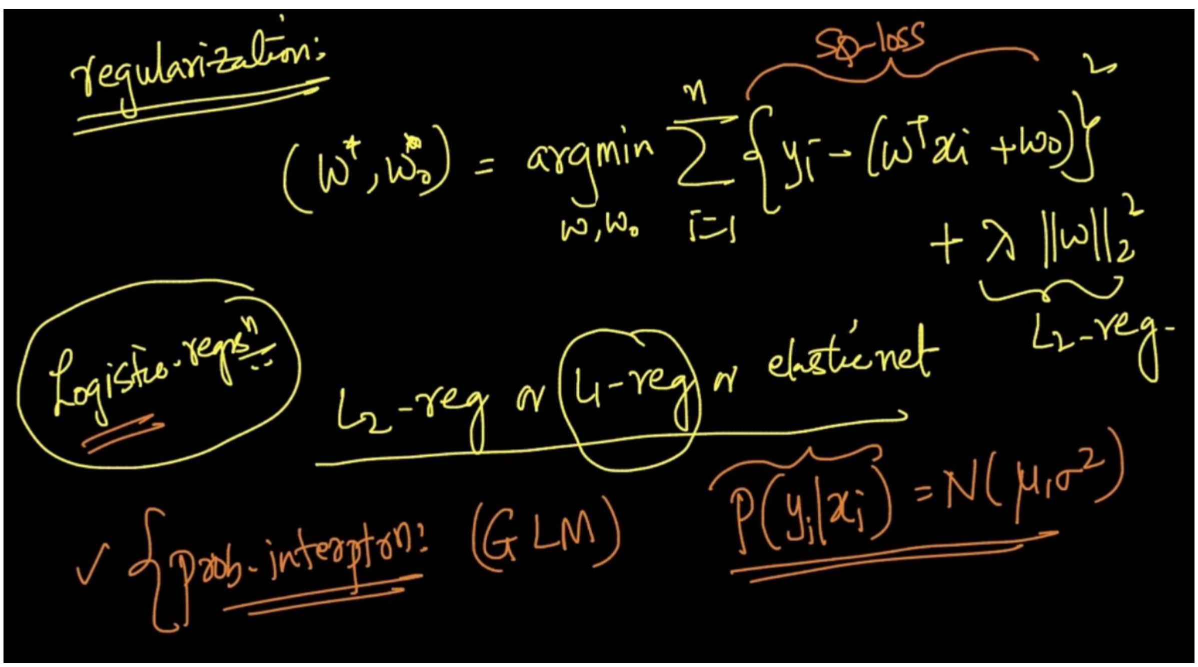
MATHEMATICAL FORMULATION



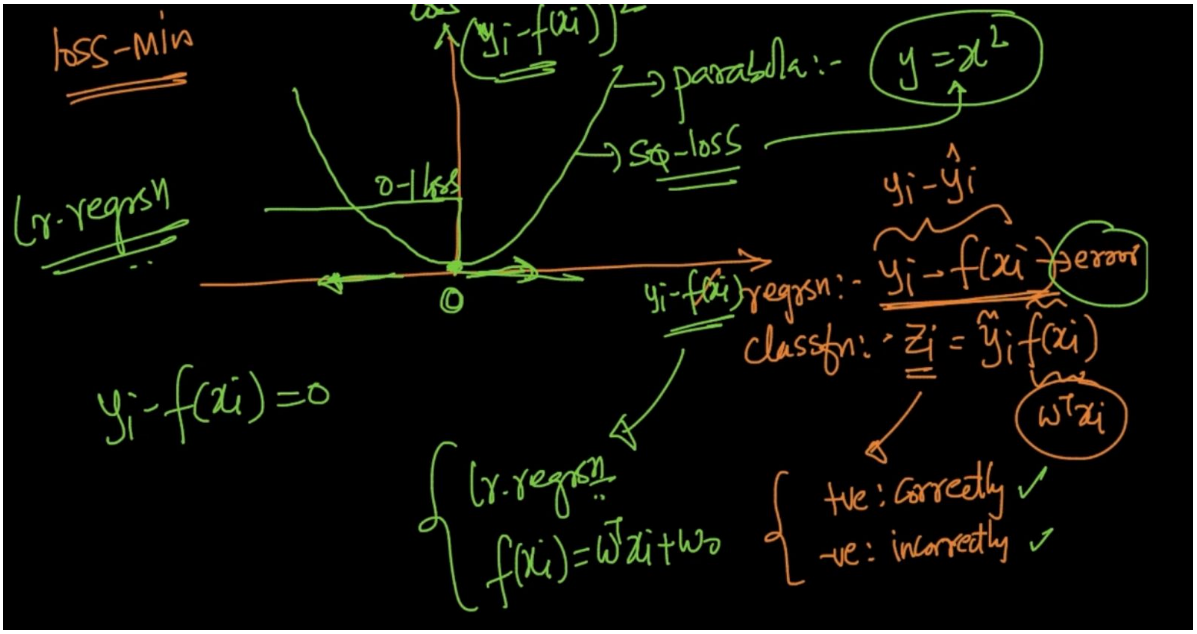
$\widehat{y}$ is our predicted value and y is the actual value and their difference is erroe. As seen it can be +ve or -ve so we'll square them up as we want the minimum sum of errors.That would be our best fit line



We want to minimize our w, w_0 . In our optimization problem we want to minimize the squared error (y - $w^T x$ + w_o) ^2 . That's our linear regression

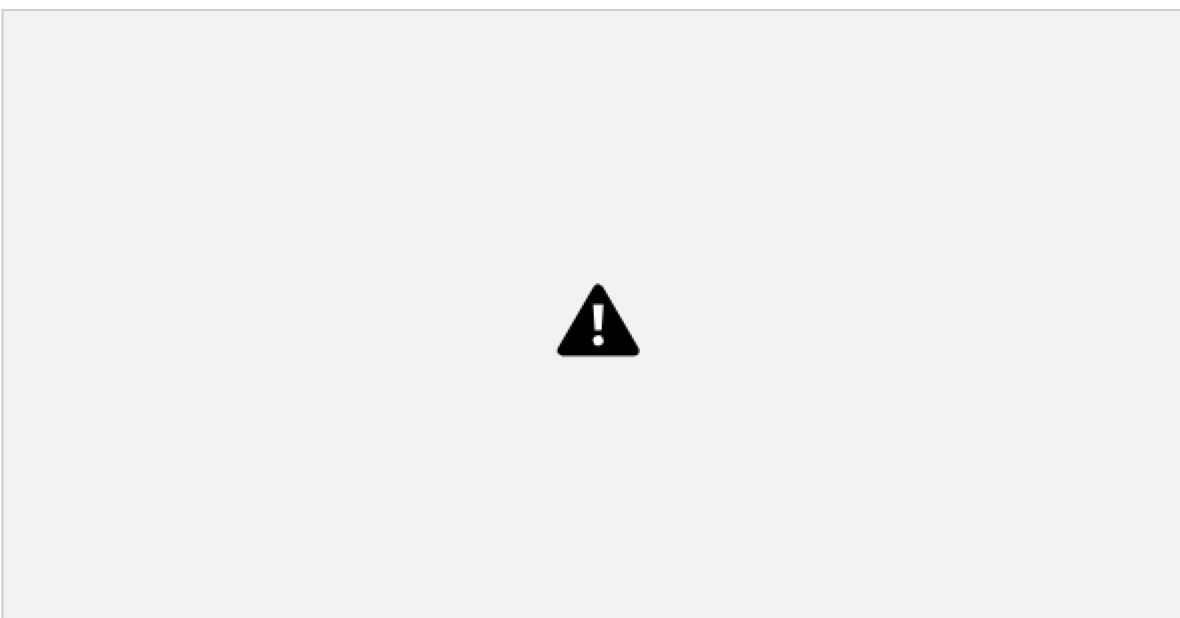


We are adding Regularization to our Linear Regression

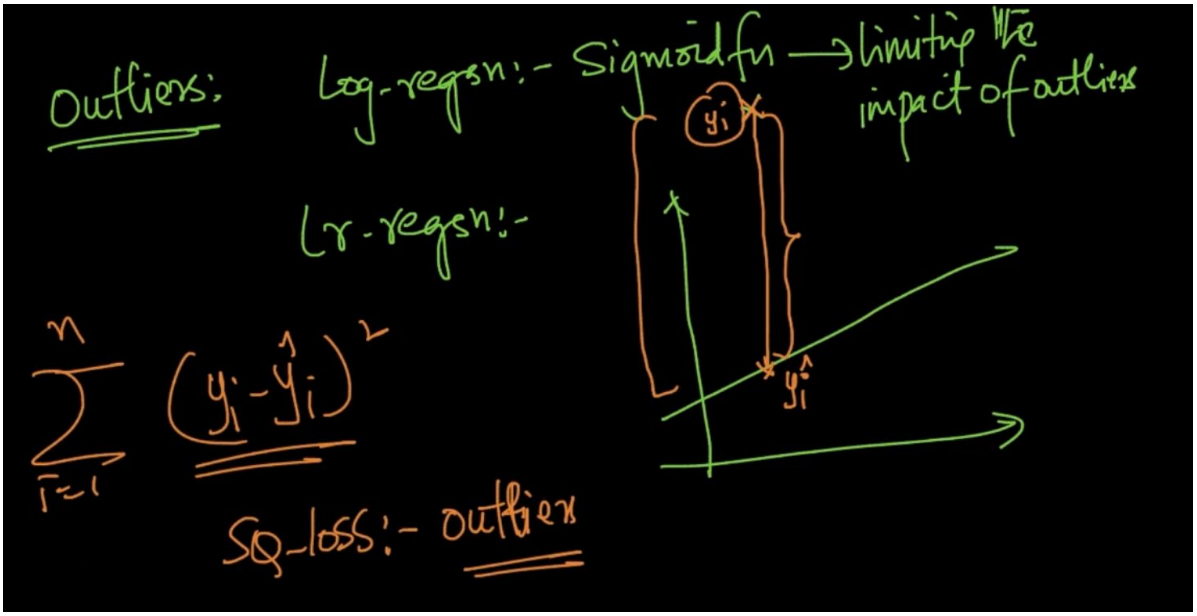


In Linear Regression, in our loss-minimization graph our x-axis = y - f(x) ,y-axis = $(y - f(x))^2$ It's a parabolic function .If the error is 0 then the loss is 0 but as the data point moves away from the hyperplane the loss increases

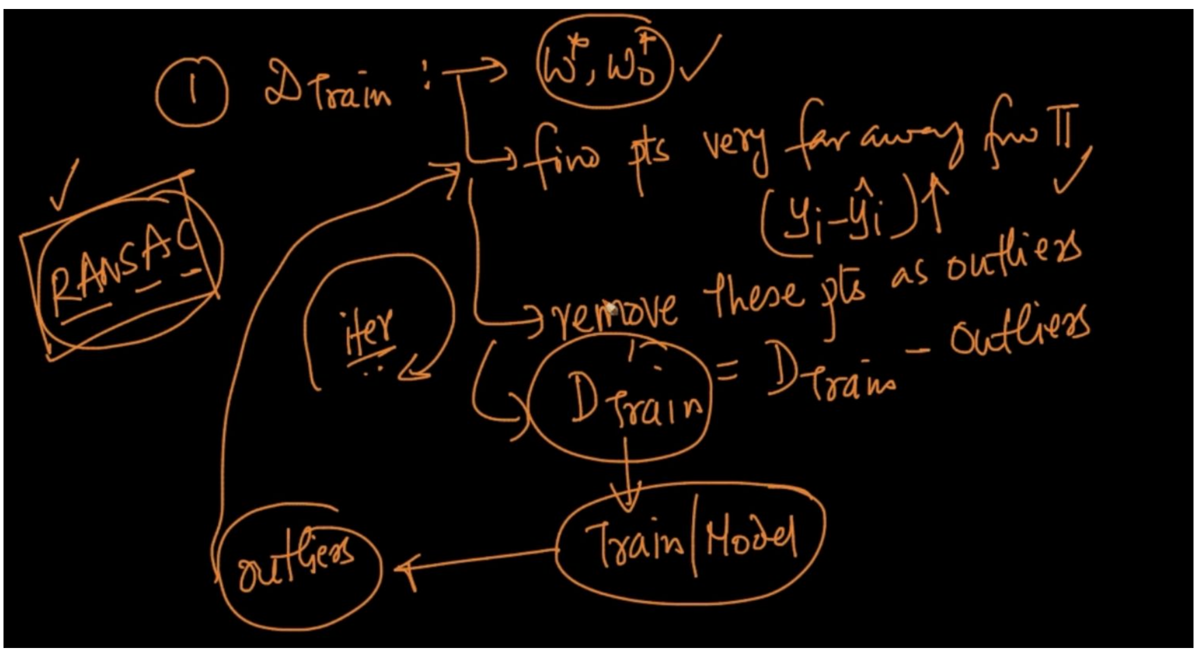
REAL WORLD CASES



All the things are done in the same way as Logistic Regression as seen above



If an outlier occurs, as seen the predicted label $\hat{y}$ and the actual value y. The difference between them is huge and we apply squared loss for optimal w , So the squared loss would create a massive effect in our y



So we find our optimal w. Find points that are very far away from (y - $\hat{y}$) and remove these points and create a new training set. Repeat the process to deal with outliers