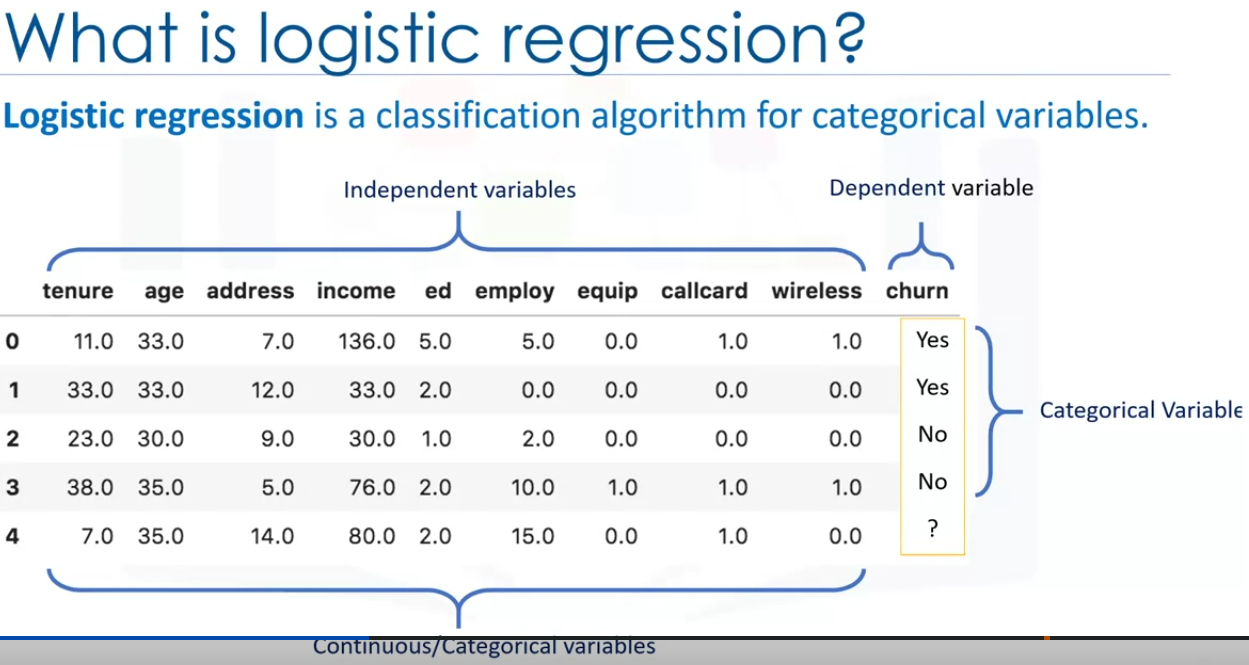
Intro of Logistic Regression



Logistic regression has a similar name to Linear regression but the logistic regression tries to predict a categorical value compared to the linear regression which predicts a numeric value.

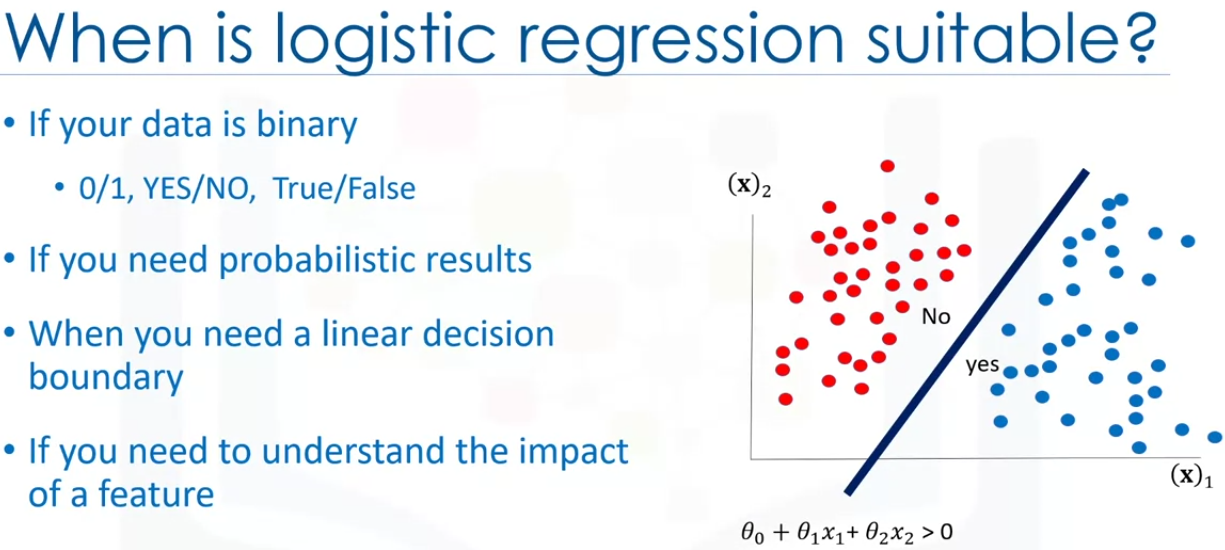
For Logistic regression to work, the independent variables needs to be continuous.

If they are categorial, they should be dummy or inidicator coded. So transormation is required.

It can be used to forbother binary and multi-class classification.



In all the above examples along with the prediction, we measure the probability .



All the data points on one side belongs to a particular category.

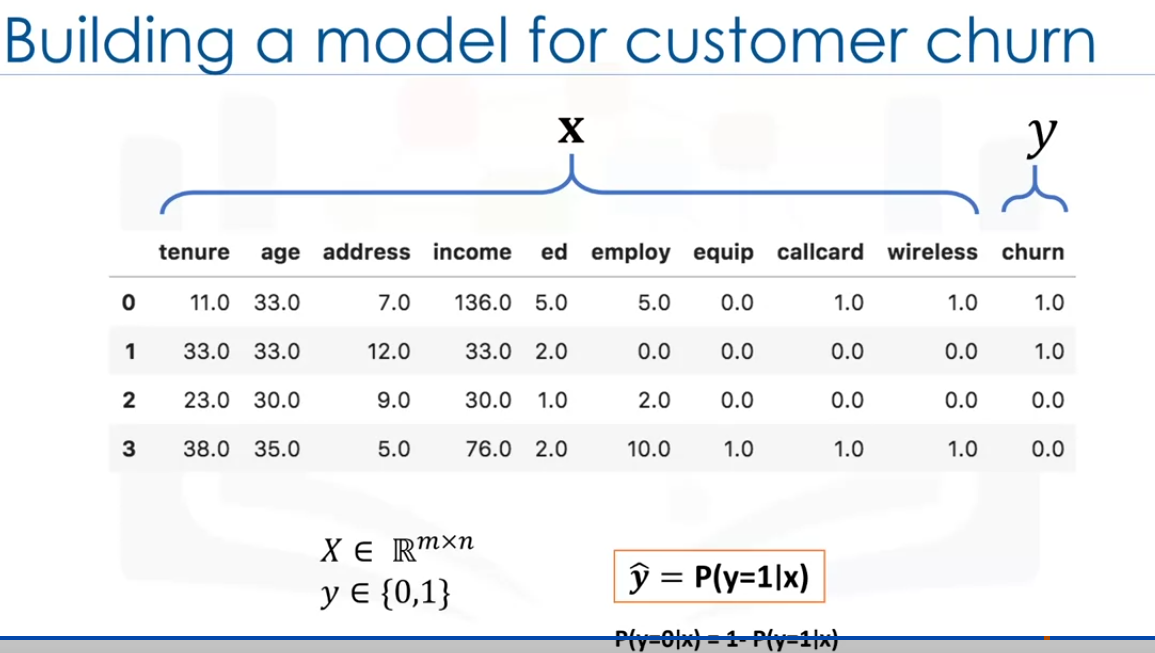
When using logistical regression, we can also achieve a complex decision boundary using polinomial processing aswell(not discussed).

More can be understood about polinomial boundaries once logistic regression is understod.

You can select the best feature based in the statistical significance of the logistic refression model’s coefficients or model parameters.

That is, after finding the optimal parameters, a feature X with the weight Theta one close to 0 hasa smaller effect on the prediction than features with lager Theta one values,

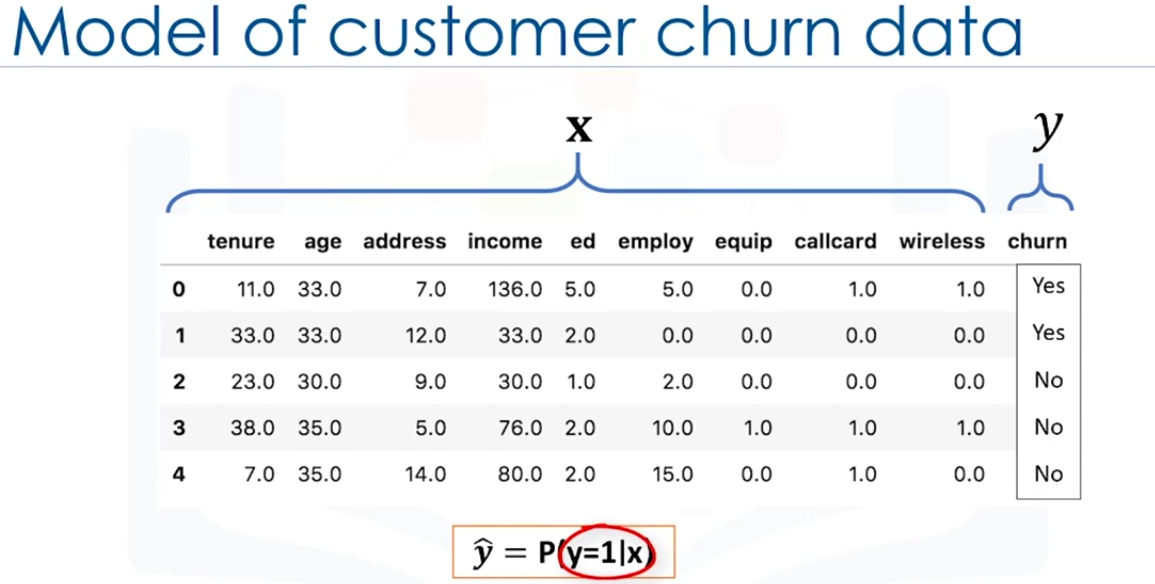
It allows us to understand an effect that an independent variable has on the dependent variable while controlling other independent variables.



For simplicity we convert the target values into 0s and 1s. The goal is the build a model to predict the class for each sample, aswell of its probability.

So we have X which is the data and y is in 0 and 1. We need to predict 0s and 1 as result.

Logical Regression Vs Linear Regression



We will lean thhe difference between linear regression and logistic regression.

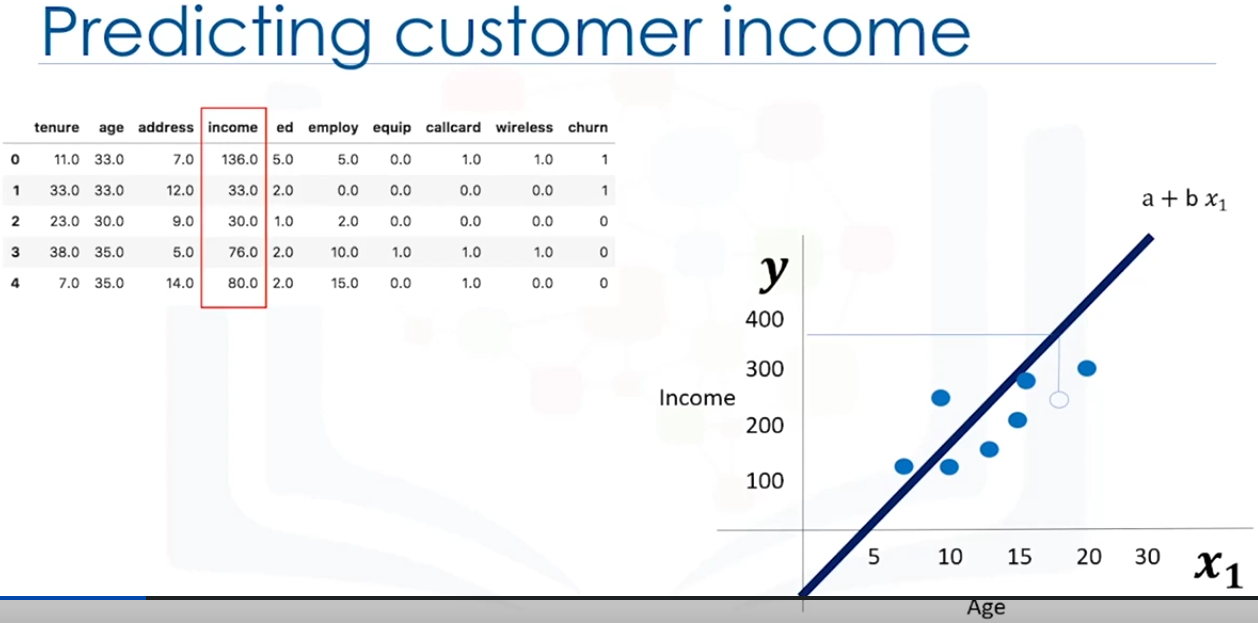
We will see how Linear Regression cannot be used for some binary classification problems.

We also look at the sigmoin function which the main part of logistic regression.

The goal of Logistic regression here is to ppredict the class of each customer and alsi the probabilty of each sample belonging to the class.

Converting the dependent column into integers, Can we solve this problem?





lets first understand how linear regression works. Lets see how to predict income.

Lets use the independent variable age and predict the dependent which is the income.

We can have more features but we are considering only age for now.

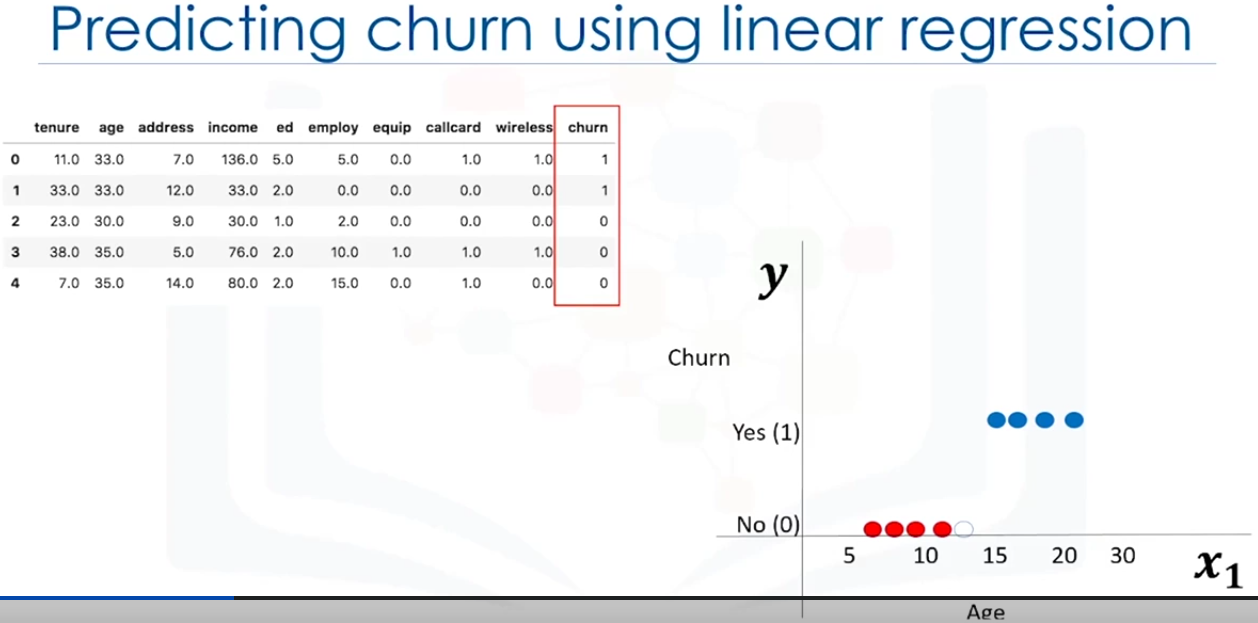
Using linear regression, we can fit a line or a polynomial though this data .We can see the example line above.

When we use the polynomial for the the line will be a quadratic equation y = mx+c where a is the m is the slope and c is the y intercept. using the values given by the LinearRegression model what we build, we can mathematically calculate the income value.

We can also plot a graph. We first plot a scatter plot for each age, income pair and then draw the line plot that best fitst those points. Alternatively, we can aslo use the regplot from sns to do this.

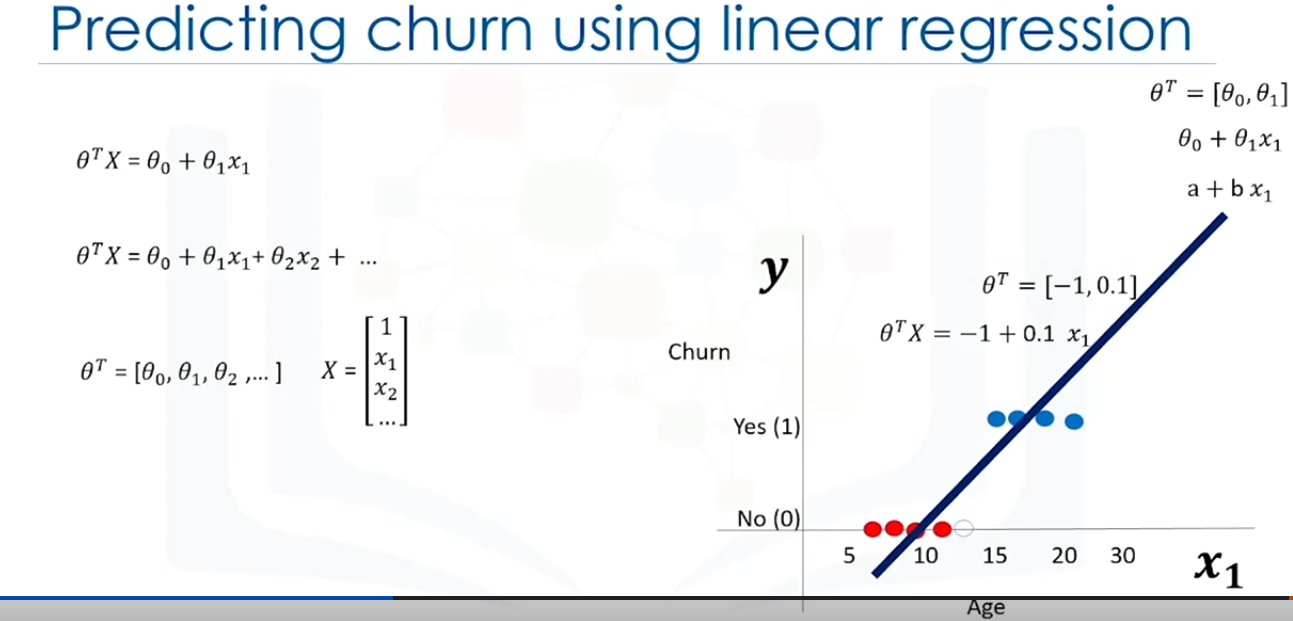
now we can we can use this line to predict the income for an age which was not seen before in the training data. We can also compare the predicted income with actual income.

What if we need to predict a categorial field like churn? Can we use the Linear regression to do this?



lets say we need to predict customer churn based on the age column, when we use the scatter plot the graphs looks like above.

We need to build a model that if the churn is yes or no when an un known age is provided.



We can use the linear regression to fit a polynomial through the data which is in the form of y= mx + c where m is slope and y intercept. it can also be written as theta1 and theta0 respectively.

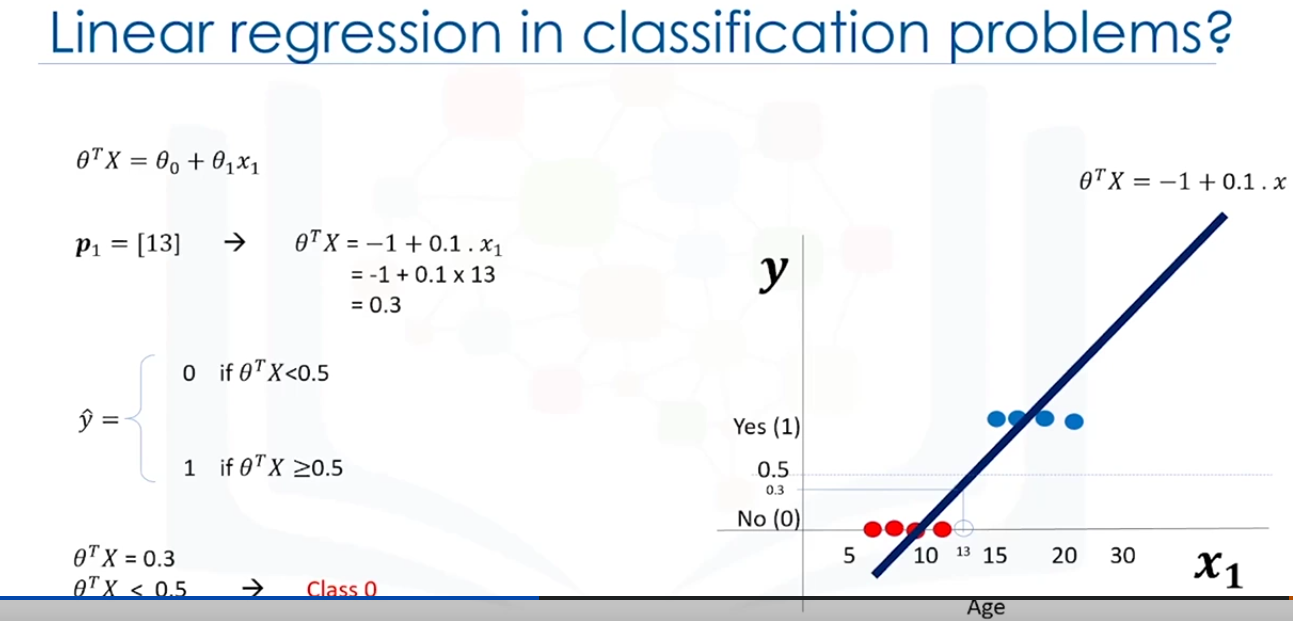
This line has 2 parameter theta0 and theta 1 represented as vector theta. we can also show this equation as theta transpose X which is represented in the left part of the image above. more complex equations with multi dimensional space can have more than 2 parameters As Theta is a vector and it is supposed to be myltiplied by X, thera is represented conventionally as theta transpose. It not very important to have that distinction or to get confused about theta or theta transpose.

Refer the video for better understanding.

Theta are called the weight factor or confidences of the equation with both these terms used interchangeably. X is the featureset which represents a customer.

We can use math or an optimization algorithm to get the theta values which helps to fit the line. For the example above the values are -1 for theta 0 and 0.1 for theta 1. Substituting these value into the fomula of line we get y = -1 + 0.1x.

now we can use this to figure out customer churn for new customers.



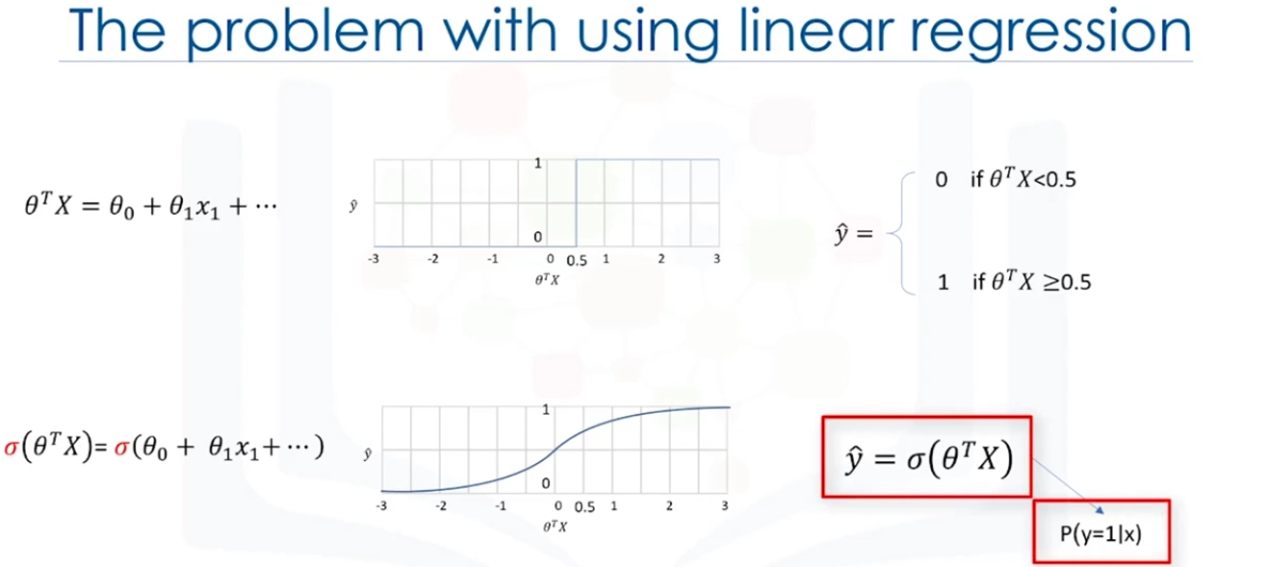
The above image calculates the churn for the customer with age = 18 which gives us 0.3. as it is less than 0.5 we put it in class 0. if the value is greater than 0.5 then it can be placed in class 1.

But there is one problem here. What is the probability that this customer belongs to class 0 ? The algorithm doent specify that.

So it is not the best model for binary classification.

Also there are some other issues which verify that tells us linear regression is not a good classification algorithm.

Read Chat gpt reponse on why Linear regression is not good for predicting binary class labels. File name : linear vs logistic regression.docx



If we use linear regression it will always give us a number such as 3 or -2 and so on . So we have to use a threshold. In this case 0.5. We can see there isa vertical line in the 1st graph at 0.5 which shows the seperation. This thresold works as a step function which is used to assign a class either 0 or 1 no matter how large or small number they are. For example 1000 will still be represented as 1 and -20000 is still represented as 0.

The line makes it vey bad judgement of and always fits it to 0 or 1 without giving any importance to the magnitude of the value.

Instead of having this step function, wouldnt it be nice if we have a smother line? One that would project these values between 0 and 1. The existing method above does not give us the probabilityof a customer belonging to a class which is very desirable.

We need the probability of a falling in a class aswell.

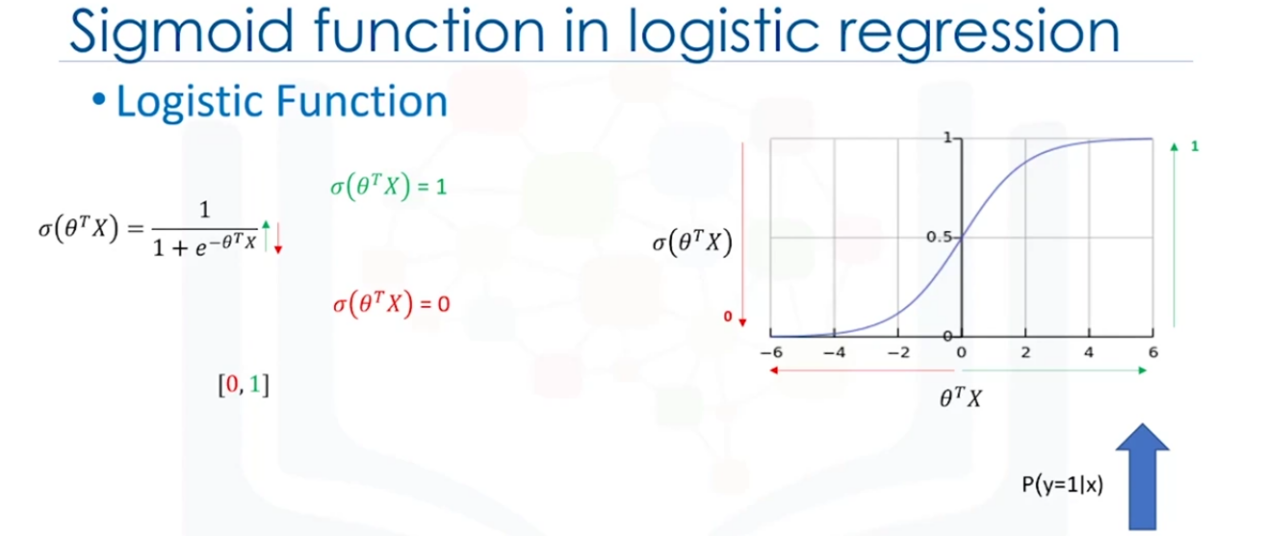
So what is the scientific solution here? Instead of using the theta transpose X, we use a very specific function called sigmoid function. The sigmoid of theta tranpose x will give us the probability of a point belonging to a class instead of the value of y directly.

It gives the probability of y being very big or small instead of giving the actual value,

It always returns a value of 0 and 1 depending on how large the theta transpose x actually is. So the formula changes to sigmoid of theta tranpose x which returns the probability of ouput is 1 given the input x.

taking this into account if we draw a graph for different values we get 2nd graph above and we can see that the lines at 0 and 1 are extending infinitely.

We have learnt that sigmoid function returns the probability but what is a sigmoid function really?

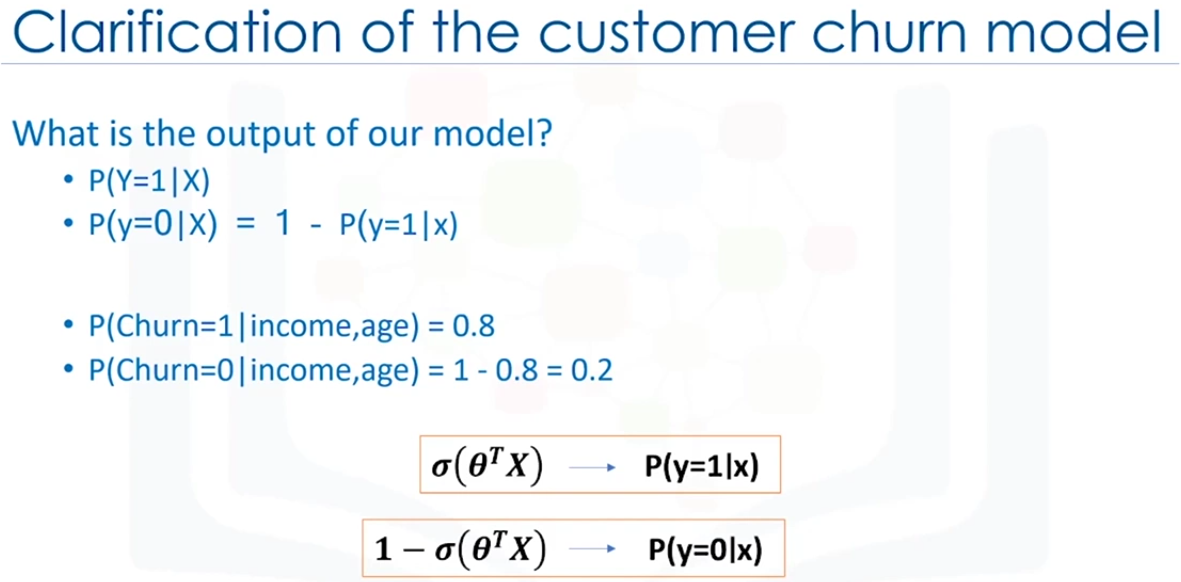


The sigmoid function is used to calculate the probability of outcome istead of giving us the class it belongs to

It is a logistic function with the formula shown in the above diagram. The formula seems a little bit scary but it will make sense once we work with it.

Notice that the negative sign for the e? if the theta transpose x becomes very larger, then the e portion of the denominator becomes close to 0 which makes the result closer to 1.

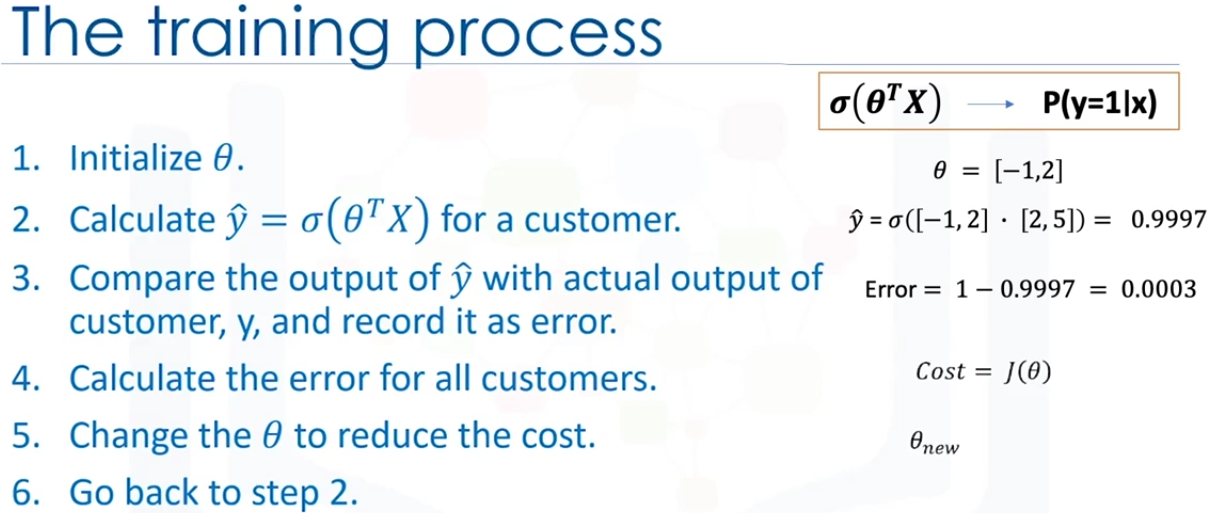
If theta transpose X is very small then the sigmoid function gets closer to 0.



the output of sigmoid function is always between 0 and 1. if it is closer to 1 then it means that the probability is high. so if the sigmoid function is closer to 1 then probability of y being 1 is very high. If the sigmoid function result is closer to 0 then the probability of y being 1 is very less.

For example, probability of y being 1 if age and income are given are is 0.8. and the probability of y being 0 is 0.2.

Now how can we achieve this. We can do this by training process.



1. Initialize the theta value with random vectors like most of the ML algorithms. In this example we take [-1, 2]
2. The compute model output which is the sigmoid of theta transpose X. In this example [2, 5 ] are the income and age respectively. The result in this case is 0.9997
3. Then we calculate the error which is 1 - sigmoid result. Here we get 0.0003
4. Calculate the error for all customers. This is represented as cost
5. Change the theta value to reduce the cost
6. Go back to step 2.

We keep doing these steps iteratively until the cost is low enough.

This bring 2 questions:

1. How can we change the theta value
2. When do we stop ?

There are many ways to change the theta value but the most popular way is to use gradient descent.

As per stopping the iterations we do that by calculating the accuracy of the model and when accuracy is satisfactory.

Logistic Regression Training

If the Content is confusing first look at the steps at the bottom.

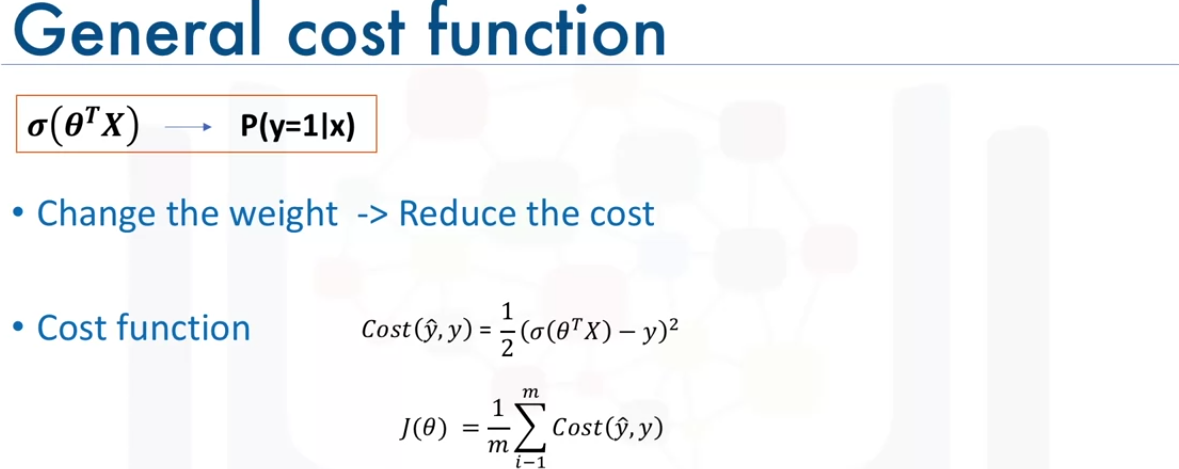
The main purpose of Logistic regression is to choose the best parameters possible to best estimation of the labels of the samples in the dataset.

To do this first we have to look at the relation between the cost function and the parameters theta.

Then using the derivative of the cost function we can find how change the parameters to reduce the cost or rather the error.

Lets see how it works.

Lets first find the cost function equation for a sample case. to do this we can use one of the customers in the churn problem.



The cost function is the difference between the actual values of y and the model output yhat. This is a general rule for most cost functions in ML.

here the yhat is the sigmoid of theta transpose x. Usually we use the square of this equation to eleminate negative values. And for simplicity half of this value is considered as the cost function for this example. Once we calculate cost for each customer, we find the mean. It is also called MSE. it is represented as J(theta).

Not how do we set the parameter to reduce the cost function?

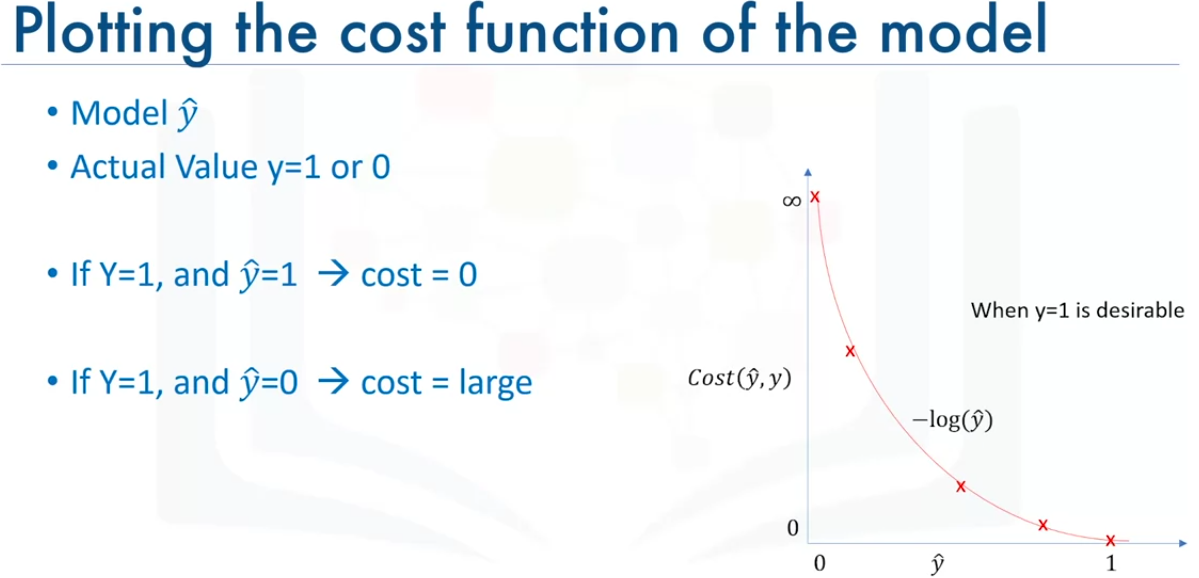
This is basically starting with a random parameters and then use gradient descent to get the best possible theta value.

Although we can find the minimum point of a function using the derivative of a function, there is not an easy way to find the global minimum point for such an equation.

Finding the global minimum is not the scope of this course.

So what can we do? We can find another cost function instead one which has the same behaviour but is easier to fund the minimum point.

Lets plot the cost function of the model.

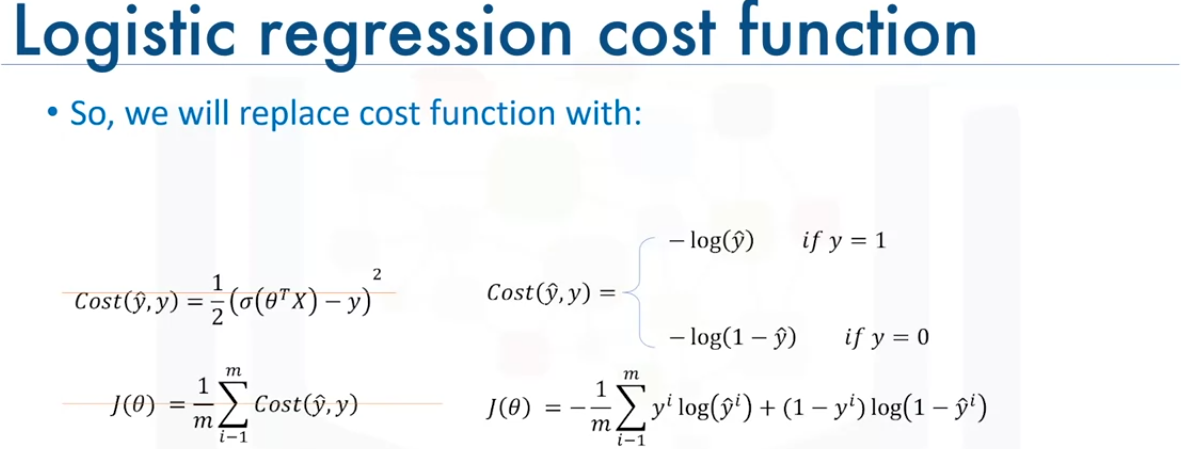


For a moment lets assume that the desired value for yhat is 1. In this case we need a cost function that returns 0 if the outcome of our model is one. Which is the same as the actual label and the cost should keep increasing as the outcome of our modek gets farther from 1 and the cost should be very large when our outcome of our model is close to 0.

We can see that the -log(yhat) gives us a const function for us. This means if the actual value and the predicted value are 1 then the -log function returns 0. But if the prediction is smaller than 1 the -log function returns a larger cost value.

So, we can use the -log function for calculating the cost of our logistic regression model.

So, we know that calculating the derivative of the cost function is difficult so we replace the const function with -log function

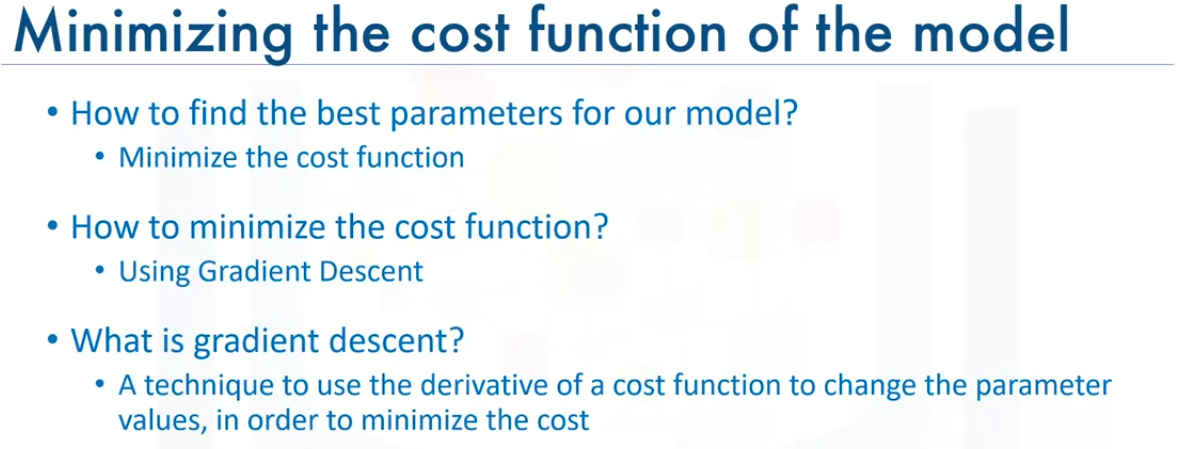


We can see here instead of using the sigmoid function we can repace it with the -log function and its mean. In cases where the desirable y should be 1 then the log function is -log(yhat) otherwise -log(1-yhat) . This can be substituted in the J(theta) and we get the above function which we use the calculate the cost.

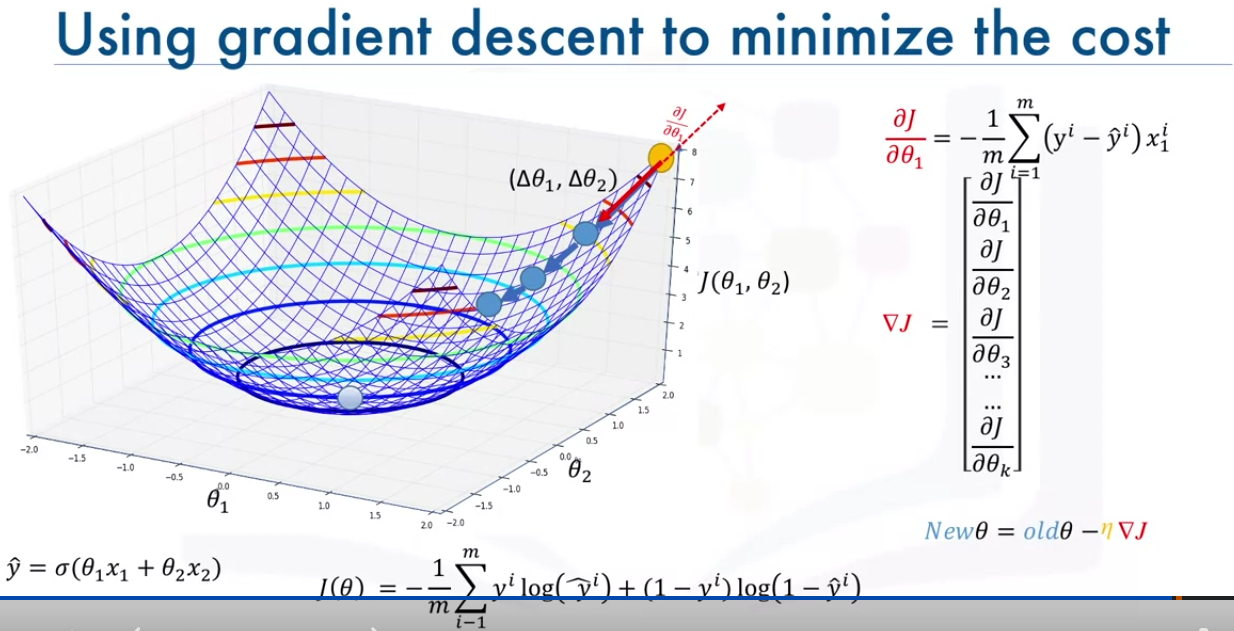
Note: the yhat is not a value of label. it is instead a probability of y being 1.

Now we can use this function to find the parameter that minimizes the cost.

Recap:



The main objective of Gradient descent is to find the parameter value by reducing cost and this is done by iteratively changing the parameters using the derivatives of the cost function.



Note: the x1 and x2 are the actual features age and income in the formula for yhat.

lets say our features are represented as a 2d plain theta1 and theta2(the bottom 2 axes) for age and income. We are ignoring the theta0 that nomally show up in the formula.

The cost function j is the mean of the -log function which we discussed in the prevous slide. We need to minimize this. So for the observed cost we add a new axes vertically which is represented as J(theta1, theta2).

When we plot a graph for the all the costs for theta1 and theta2 the above graph will be outcome. It represents different error values of parameters.

This graph is called error curve or error bowl of the cost function.

We want to use this error bowl to find the best parameter values that reult in the min cost.

Which point is the best point .

We should reduce the position on the curve. Again we need to reduce the cost.

But how ? Will we add some values to our weights or deduct some value and what is the value we have to use?

We can start by choosing a random point on the bowl. Lets say the yellow point we change the value by (delta theta1 , delt theta2) and we get the first blue point and we keep doing this as long as we go down the slope until we reach a flat surface. This will be the min point on the curve and has the optimum theta 1 and theta2, This is reprented as the point at the center of the bowl the the above graph.

But there are more questions:

which directions do we need to go?

how large should each step geing to be?

To know this we need to find the gradoent of the cost function at that point.

The gradient is the slope of the serface at every point and the direction of the gradient is the direction of the greatest uphill.

So how do we calculate the gradient of the cost function?

If we take a random point for example the yellow point, and take the patial derivative of J of theta with respect to each parameter at that point, it gives us the slope of the move for each parameter at that point.

If we move in the opposite direction of that slope, It gaarentees that we go down in the error curve.

For example if we dinf the derivative of J with respect to theta1 , it gives us a positive number. This indicates that the cost function is increasing as theta1 increases. So to decrease j, we need to move in the opposite direction. So we move towards the negative slope(theta1). We have to calculate the derivatives at each step.

The gradient value also tell how big a step to take. If the slope is large, we take a large step as we are far from the minimum and if the slope is small we take small steps. in the graph we can se the distance between each point is reducing as we go down.

As we can see the formula the partial derivative of J over theta 1 where the we find the negative mean of the cost functions (y-yhat)x . So instead of applying the derivative we can use the mean formula. In a nutshell it returns the slope of that point and we should update the parameter in the opposite direction of the slope.

A vector of these slopes is the gradient vector. We can use this vector to change of update the parameter we take the current parameters and subtract the error derivative(newTheta = oldTheta - deltaJ) as theta is a vector of all the parameters. We can be sure that the cost of the theta will be reduced by this. Also we multiply a constant value to gradient(deltaJ) which is called mu which is called the learning rate.

Learning rate gives us additional control on how fast we move on the surface.

In other words gradient descent is the direction of the step and the Learning rate is the length of the step.

This happens iteratively and reducing the cost.

To put it simply, when applying gradient descent with the logistic regression to find the best parameters, the algorithm does the following things.

1. Takes a random theta values.
2. Then finds the yhat (which is a probability) using the sigmoid function(yhat = sig(thetaT X)). Sigmoid function calculates the probability based on the actual predicted value.
3. Then the algorithm uses the probability and compares it with the actual value to find the error for a single customer using -log function(-log(yhat) or -log(1-yhat) based on if the predicted value is 1 or 0) which is just and alternative for (1/2) \*(yhat - y)2.
4. We calculate the same for all the customers and find the mean which is represented as J which is also called log loss cost function as we are using log loss here to calulate the cost at each customer.
5. Then we find the deltaJ which the gradient derivatives of J and subtract them form theta to getnew theta values.
6. We repeat this till we a small enough cost and then use the that to create a new model and use it for prediction.

