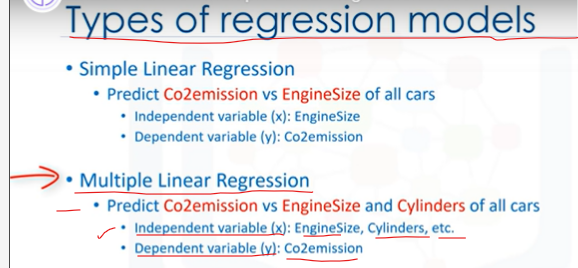
**2nd Part of this Module: MLR (Multiple Linear regression)**

**We’ll be covering multiple linear regression.**

**As you know there are two types of linear regression models: simple regression and multiple**



1. Simple linear regression is when one independent variable is used to estimate a dependent variable. For example, predicting Co2 emission using the variable of EngineSize.

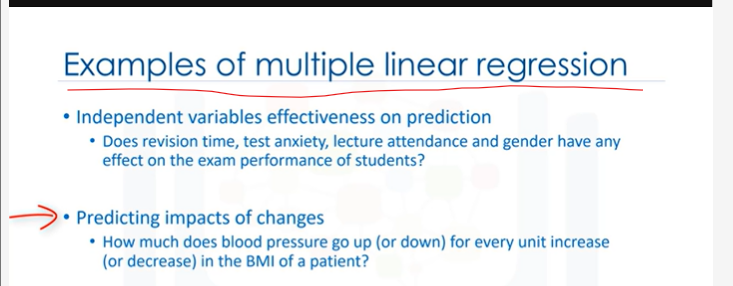


1. In reality, there are multiple variables that predict the Co2 emission. When multiple independent variables are present, the process is called "multiple linear regression." For example, predicting Co2 emission using EngineSize and the number of Cylinders in the car’s engine.
2. Our focus in this video is on multiple linear regression, The good thing is that multiple linear regression is the extension of the simple linear regression model.

So, I suggest you go through the Simple Linear Regression video first, if you haven’t **watched it already!**

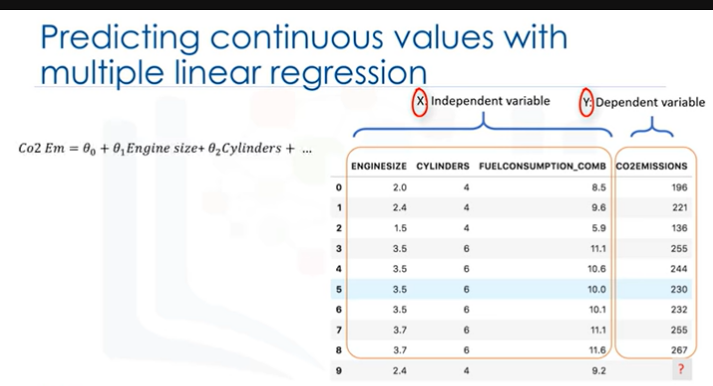
**Before we dive into a sample dataset and see how multiple linear regression works, I want to tell you what kind of problems it can solve; when we should use it; and, specifically, what kind of questions we can answer using it.**

**Basically, there are two applications for multiple linear regression.**



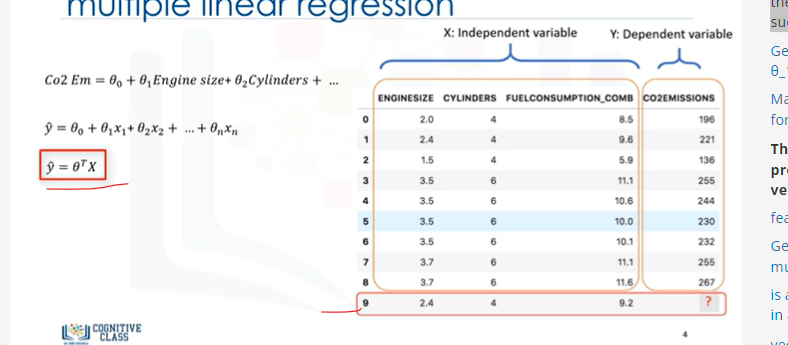
1. First, it can be used when we would like to identify the strength of the effect that the independent variables have on a dependent variable.
2. For example, does Revision time, Test anxiety, Lecture attendance, and Gender, have any effect on EXAM Performance of students?
3. Second, it can be used to predict the impact of changes. That is, to understand how the dependent variable changes when we change the independent variables.
4. For example, if we were reviewing a person’s health data, a multiple linear regression can tell you how much that person’s blood pressure goes up (or down) for every unit increase (or decrease) in a patient’s body mass index (BMI), holding other factors constant.

As is the case with simple linear regression, multiple linear regression is a method of **predicting a continuous variable. It uses multiple variables, called independent** variables, or predictors, that best predict the value of the target variable, which is also called the dependent variable



1. In multiple linear regression, the target value, y, is a linear combination of independent variables, x.
2. For example, you can predict how much Co2 a car might emit due to independent variables, **such as the car’s Engine Size, Number of Cylinders and Fuel Consumption.**
3. **Multiple linear regression is very useful because you can examine which variables are significant predictors of the outcome variable. Also, you can find out how each feature impacts the outcome variable!**

And again, as is the case in simple linear regression, if you manage to build such a regression model, you can use it to predict the emission amount of an unknown case, such as record number 9.



**Y = Θ.o + Θ.1x (This is Simple Linear regression!)**

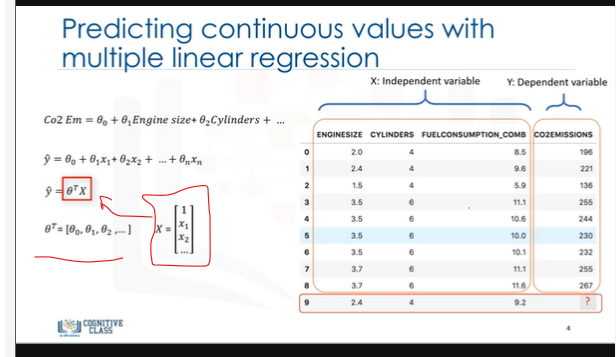
**θ1 is known as the "slope" or "gradient" of the fitting line and θ0 is known as the "intercept."**

1. **Generally, the model is of the form:**

**y ̂ = θ0+ θ1 x1+ θ2 x2 and so on, up to ... +θ\_n x\_n.**

1. Mathematically, we can show it as a vector form as well.
2. This means, it can be shown as a dot product of 2 vectors:

the parameters vector and the feature set vector.



1. Generally, we can show the equation for a multi-dimensional space as

θ^T\*X, where θ is an n-by-one vector of unknown parameters in a multi-dimensional space,

and x is the vector of the feature sets, as θ is a vector of coefficients, and is supposed to be multiplied by X.

Conventionally, it is shown as transpose θ.

1. θ is also called the parameters, or, weight vector of the regression equation … both these terms can be used interchangeably. And X is the feature set, which represents a car.

For example x1 for engine size, or x2 for cylinders, and so on.

1. The first element of the feature set would be set to 1, because it turns the θ0 into the intercept or bias parameter when the vector is multiplied by the parameter vector.
2. Please notice that θ^T x in a one-dimensional space, is the equation of a line. It is what we use in simple linear regression.
3. **In higher dimensions, when we have more than one input (or x), the line is called a plane or a hyper-plane. And this is what we use for Multiple Linear Regression.**

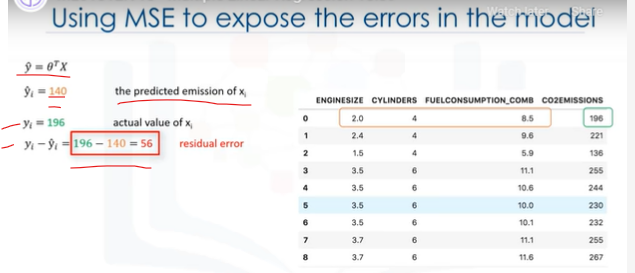
**So, the whole idea is to find the best fit hyper-plane for our data.**

1. To this end, and as is the case in linear regression, we should estimate the values for θ vector that best predict the value of the target field in each row.
2. To achieve this goal, we have to minimize the error of the prediction.

**Now, the question is, "How do we find the optimized parameters?"**

To find the optimized parameters for our model, we should first understand what the optimized parameters are. Then we will find a way to optimize the parameters.

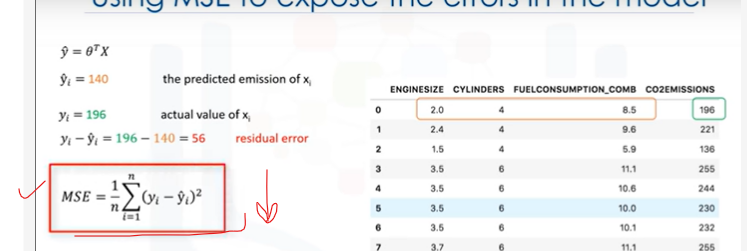
1. In short, optimized parameters are the ones which lead to a model with the fewest errors.
2. Let’s assume, for a moment, that we have already found the parameter vector of our model. It means we already know the values of θ vector.
3. Now, we can use the model, and the feature set of the first row of our dataset to predict the Co2 emission for the first car, correct?
4. **If we plug the feature set values into the model equation, we find y ̂ .**
5. Let’s say, for example, it returns 140 as the predicted value for this specific row.
6. What is the actual value? y=196. How different is the predicted value from the actual value of 196? Well, we can calculate it quite simply, as 196-140, which of course = 56. This is the error of our model, only for one row, or one car, in our case



1. **As is the case in linear regression, we can say the error here is the distance from the data point to the fitted regression model.**

**So, what is MSE come to Picture Here?**

1. The mean of all residual errors shows how bad the model is representing the dataset. It is called the mean squared error, or MSE.
2. Mathematically, MSE can be shown by an equation. While this is not the only way to expose the error of a Multiple linear regression model, it is one the most popular ways to do so.
3. The best model for our dataset is the one with minimum error for all prediction values. So, the objective of multiple linear regression is to minimize the MSE equation.

To minimize it, we should find the best parameters θ, but how?

**Que) “How do we find the parameter or coefficients for multiple linear regression?”**

* There are many ways to estimate the value of these coefficients.
* However, the most common methods are the ordinary least squares and Optimization approach.
* Ordinary least squares(OLS) tries to estimate the values of the coefficients by minimizing the “Mean Square Error.” This approach uses the data as a matrix and uses Linear Algebra operations to estimate the optimal values for the theta.
* The problem with this OLS technique is the time complexity of calculating matrix operations, as it can take a very long time to finish.
* When the number of rows in your dataset is less 10,000 you can think of this technique as an option,
* However, for greater values, you should try other faster approaches.
* The Second option is to use an Optimization Algorithm to find the best parameters.
* That is, you can use a process of optimizing the values of the coefficients by iteratively minimizing the error of the model on your training data.

**Here Gradient Descent Method can be used?**

1. For example, you can use Gradient Descent, which starts optimization with random values for each coefficient.
2. Then, calculates the errors, and tries to minimize it through wise changing of the coefficients in multiple iterations.
3. Gradient descent is a proper approach if you have a large dataset.
4. Please understand, however, that there are other approaches to estimate the parameters of the multiple linear regression that you can explore on your own.

**Above Recap!**



**After you find the best parameters for your model, you can go to the prediction phase.**

1. After we found the parameters of the linear equation, making predictions is as simple as solving the equation for a specific set of inputs.

Imagine we are predicting Co2 emission (or y) from other variables for the automobile in record number 9.

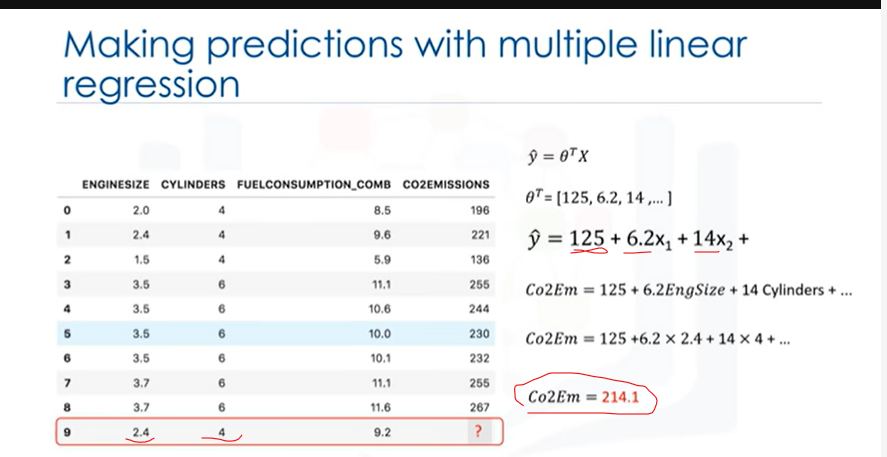
* Our linear regression model representation for this problem would be: y ̂=θ^T x. Once we find the parameters, we can plug them into the equation of the linear model.
* For example, let’s use θ0 = 125, θ1 = 6.2, θ2 = 14, and so on.
* If we map it to our dataset, we can rewrite the linear model as

"Co2Emission=125 plus 6.2 multiplied by EngineSize plus 14 multiplied by Cylinder," and so on.

* As you can see, multiple linear regression estimates the relative importance of predictors.
* For example, it shows Cylinder has higher impact on Co2 emission amounts in comparison with EngineSize.

1. Now, let’s plug in the 9th row of our dataset and calculate the Co2 emission for a car with the EngineSize of 2.4.
2. So Co2Emission=125 + 6.2 × 2.4 + 14 × 4 … and so on.

We can predict the Co2 emission for this specific car would be 214.1.



**Now let me address some concerns that you might already be having regarding multiple linear regression.**

1. As you saw, you can use multiple independent variables to predict a target value in multiple linear regression.
2. It sometimes results in a better model compared to using a simple linear regression, which uses only one independent variable to predict the dependent variable.

**How to Determine whether to use SLR or MLR?**

**Now, the question is, "How many independent variables should we use for the prediction?”**

**Should we use all the fields in our dataset? Does adding independent variables to a multiple linear regression model always increase the accuracy of the model?**

* Basically, adding too many independent variables without any theoretical justification may result in an over-fit model.
* An over-fit model is a real problem because it is too complicated for your data set and not general enough to be used for prediction.
* So, it is recommended to avoid using many variables for prediction.
* There are different ways to avoid overfitting a model in regression, however, that is outside the scope of this video.

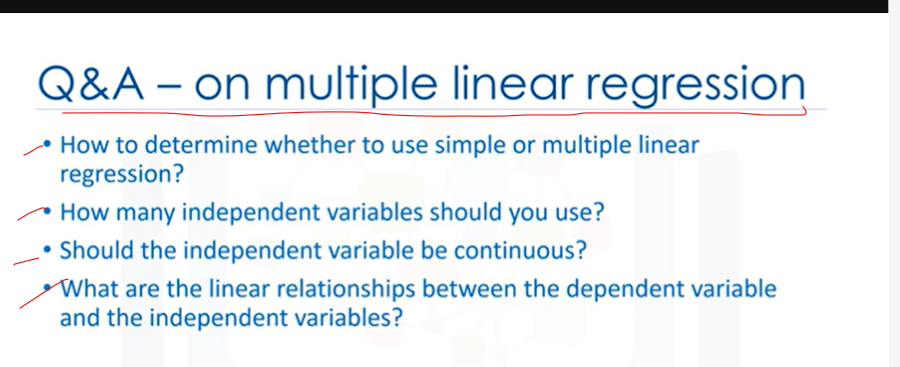
**The next question is, “Should independent variables be continuous?”**

* Basically, categorical independent variables can be incorporated into a regression model by converting them into numerical variables.
* For example, given a binary variable such as car type, the code dummies “0” for “Manual” and 1 for “automatic” cars.

**What are the linear relationship between Dependent and Independent Variable?**

1. As a last point, remember that “multiple linear regression” is a specific type of linear regression. So, there needs to be a linear relationship between the dependent variable and each of your independent variables.
2. There are a number of ways to check for linear relationship.
3. For example, you can use scatterplots, and then visually check for linearity.
4. If the relationship displayed in your scatterplot is not linear, then, you need to use non-linear regression.

**RFECAP of Above 4 Questions!**



This concludes our video. Thanks for watching.

NEXT Video:

**Model Evaluation in Regression Models (8:27)**

We’ll be covering model evaluation.

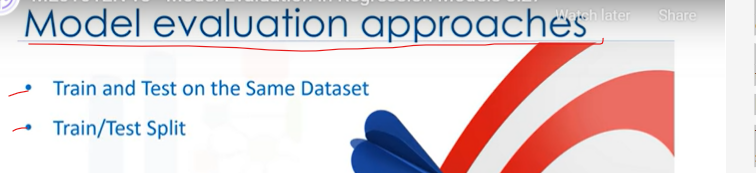


The goal of regression is to build a model to accurately predict an unknown case.

1. To this end, we have to perform regression evaluation after building the model.

In this video, we’ll introduce and discuss two types of evaluation approaches that can be used to achieve this goal.

**These approaches are: train and test on the same dataset, and train/test split.**

****

We’ll talk about what each of these are, as well as the pros and cons of using each of these models.

1. Also, we’ll introduce some metrics for accuracy of regression models.

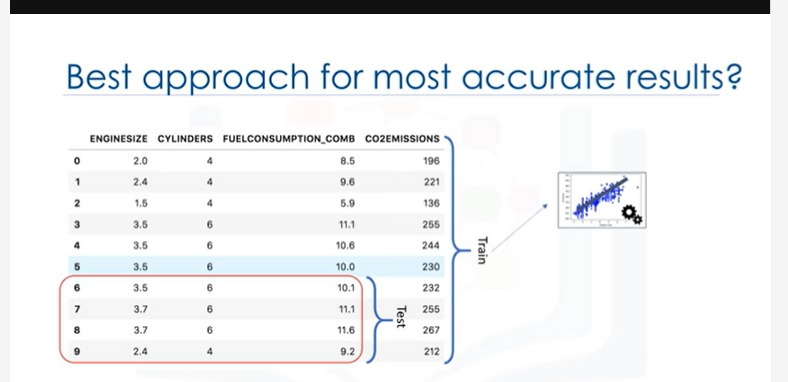
Let’s look at the first approach.

1. When considering evaluation models, we clearly want to choose the one that will give us the most accurate results.

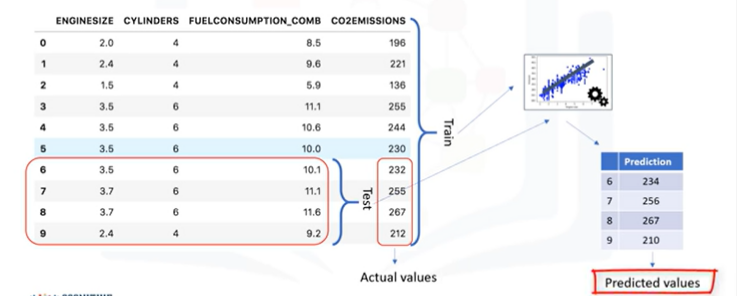
**So, the question is, how we can calculate the accuracy of our model?**

In other words, how much can we trust this model for prediction of an unknown sample,

1. using a given a dataset and having built a model such as linear regression.
2. One of the solutions is to select a portion of our dataset for testing.
3. For instance, assume that we have 10 records in our dataset.
4. We use the entire dataset for training, and we build a model using this training set.
5. Now, we select a small portion of the dataset, such as row numbers 6 to 9, but without the labels.
6. This set, is called a test set, which has the labels, but the labels are not used for prediction, and is used only as ground truth.
7. The labels are called “Actual values” of the test set.

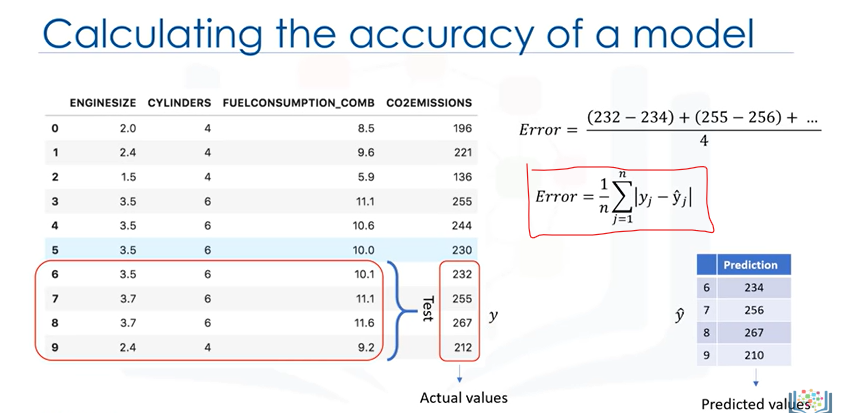


1. Now, we pass the feature set of the testing portion to our built model, and predict the target values.
2. Finally, we compare the predicted values by our model with the actual values in the test set. This indicates how accurate our model actually is!



1. There are different metrics to report the accuracy of the model, but most of them work generally, based on the similarity of the predicted and actual values.

Let’s look at one of the simplest metrics to calculate the accuracy of our regression model.

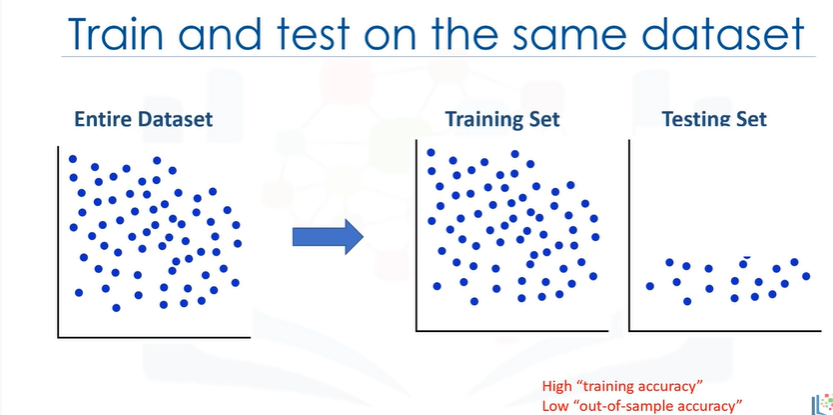


1. As mentioned, we just compare the actual values, y, with the predicted values, which is noted as y ̂ for the testing set.
2. The error of the model is calculated as the average difference between the predicted and actual values for all the rows.
3. We can write this error as an equation.

**So, the first evaluation approach we just talked about is the simplest one:**

* train and test on the SAME dataset.

1. Essentially, the name of this approach says it all … you train the model on the entire dataset, then you test it using a portion of the same dataset.
2. In a general sense, when you test with a dataset in which you know the target value for each data point, you’re able to obtain a percentage of accurate predictions for the model.
3. This evaluation approach would most likely have a high “training accuracy” and a low “out-of-sample accuracy”, since the model knows all of the testing data points from the training.



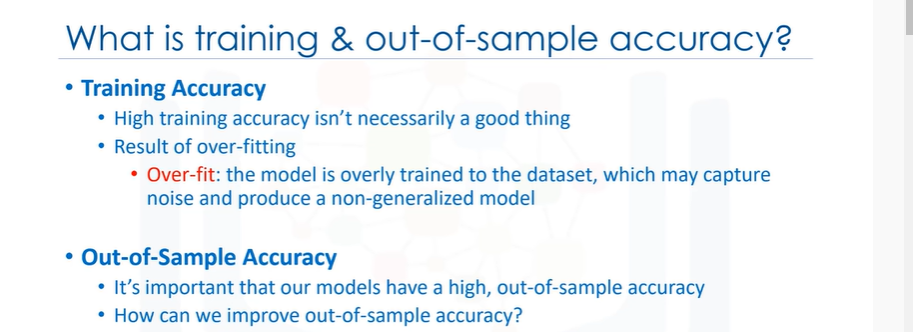
**What is training accuracy and Out-of-sample accuracy?**

1. **We said that training and testing on the same dataset produces a high training accuracy,** but **what exactly is "training accuracy?"**

* Training accuracy is the percentage of correct predictions that the model makes when using the test dataset.

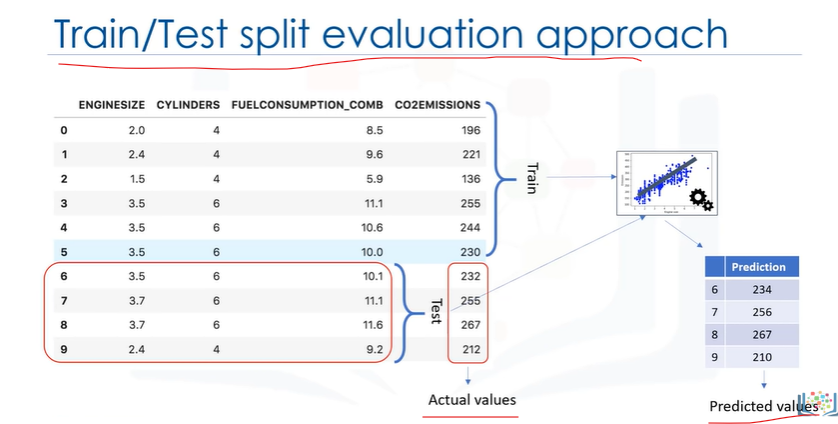
1. However, a high training accuracy isn’t necessarily a good thing.
2. For instance, having a high training accuracy may result in an ‘over-fit’ of the data.
3. This means that the model is overly trained to the dataset, which may capture noise and produce a non-generalized model.
4. Out-of-sample accuracy is the percentage of correct predictions that the model makes on data that the model has NOT been trained on.
5. **Doing a “train and test” on the same dataset will most likely have low out-of-sample accuracy** due to the likelihood of being over-fit.
6. It’s important that our models have high, out-of-sample accuracy, because the purpose of our model is, of course, to make correct predictions on unknown data.

**Above Recap!**



**So, how can we improve out-of-sample accuracy?**

1. One way is to use another evaluation approach called "Train/Test Split."
2. In this approach, we select a portion of our dataset for training, for example, rows 0 to 5. And the rest is used for testing, for example, rows 6 to 9.
3. The model is built on the training set.
4. Then, the test feature set is passed to the model for prediction.
5. And finally, the predicted values for the test set are compared with the actual values of the testing set.



**This second evaluation approach, is called "Train/Test Split."**

1. Train/Test Split involves splitting the dataset into training and testing sets, respectively, which are mutually exclusive, after which, you train with the training set and test with the testing set.
2. This will provide a more accurate evaluation on out-of-sample accuracy because the testing dataset is NOT part of the dataset that has been used to train the data.
3. It is more realistic for real world problems.
4. This means that we know the outcome of each data point in this dataset, making it great to test with!

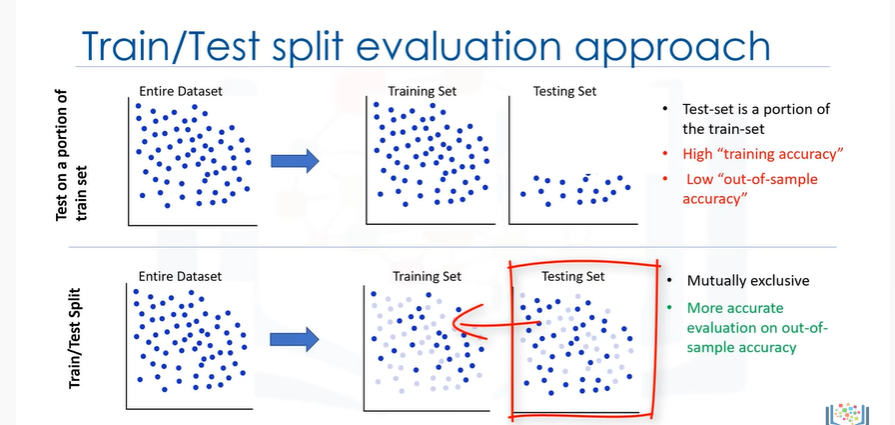
And since this data has not been used to train the model, the model has no knowledge of the outcome of these data points.

So, in essence, it’s truly out-of-sample testing.

1. However, please ensure that you train your model with the testing set afterwards, as you don’t want to lose potentially valuable data.

The issue with train/test split is that it’s highly dependent on the datasets on which the data was trained and tested.

1. The variation of this causes train/test split to have a better out-of-sample prediction than training and testing on the same dataset, but it still has some problems due to this dependency.



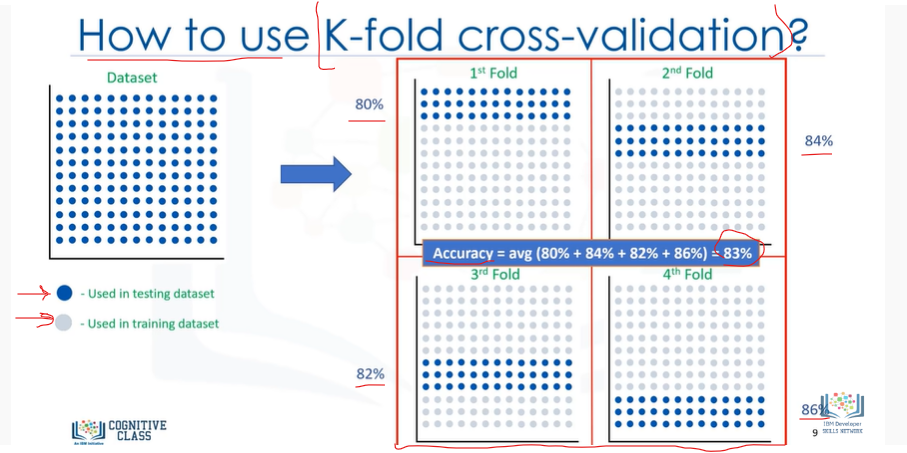


Another evaluation model, called **"K-Fold Cross-validation,"** resolves most of these issues.

**How do you fix a high variation that results from a dependency?**

Well, you average it.

1. Let me explain the basic concept of “k-fold cross-validation” to see how we can solve this problem.
2. The entire dataset is represented by the points in the image at the top left.
3. If we have k=4 folds, then we split up this dataset as shown here.
4. In the first fold, for example, we use the first 25 percent of the dataset for testing, and the rest for training.
5. The model is built using the training set, and is evaluated using the test set.
6. Then, in the next round (or in the second fold), the second 25 percent of the dataset is used for testing and the rest for training the model.
7. Again the accuracy of the model is calculated.
8. We continue for all folds.
9. Finally, the result of all 4 evaluations are averaged.



That is, the accuracy of each fold is then averaged, keeping in mind that each fold is distinct, where no training data in one fold is used in another.

1. K-fold cross-validation, in its simplest form, performs multiple train/test splits using the same dataset where each split is different.

Then, the result is averaged to produce a more consistent out-of-sample accuracy.

We wanted to show you an evaluation model that addressed some of the issues we’ve described in the previous approaches.

However, going in-depth with the K-fold cross-validation model is out of the scope for this course.

Thanks for watching!

**NEXT Video:**

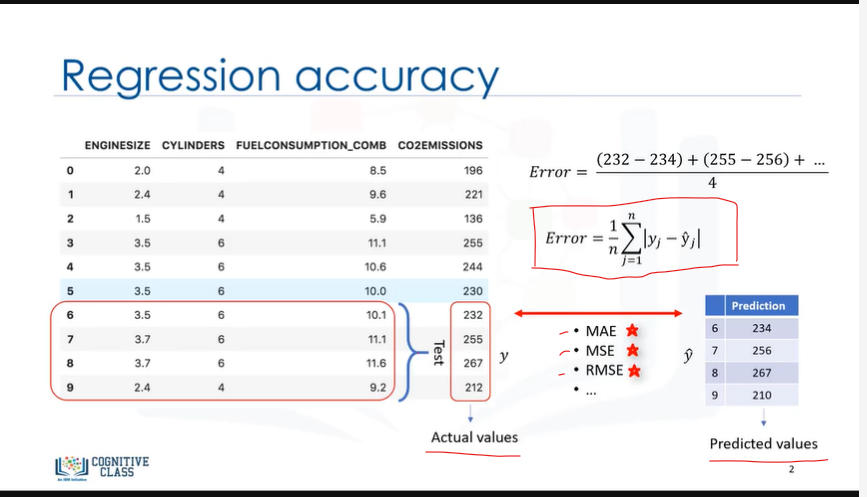
## Evaluation Metrics in Regression (3:06)



**we’ll be covering accuracy metrics for model evaluation.**

1. Evaluation metrics are used to explain the performance of a model.
2. Let’s talk more about the model evaluation metrics that are used for regression.
3. As mentioned, basically, we can compare the actual values and predicted values to calculate the accuracy of a regression model.
4. Evaluation metrics provide a key role in the development of a model, as it provides insight to areas that require improvement.
5. We’ll be reviewing a number of model evaluation metrics, including:

**Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE).**

****

**But, before we get into defining these, we need to define what an error actually is.**

1. In the context of regression, the error of the model is the difference between the data points and the trend line generated by the algorithm.
2. Since there are multiple data points, an error can be determined in multiple ways.

**1)Mean absolute error is the mean of the absolute value of the errors.**

1. This is the easiest of the metrics to understand, since it’s just the average error.

**2)Mean Squared Error (MSE) is the mean of the squared error.**

1. It’s more popular than Mean absolute error because the focus is geared more towards large errors.
2. This is due to the squared term exponentially increasing larger errors in comparison to smaller ones.

**3)Root Mean Squared Error (RMSE) is the square**

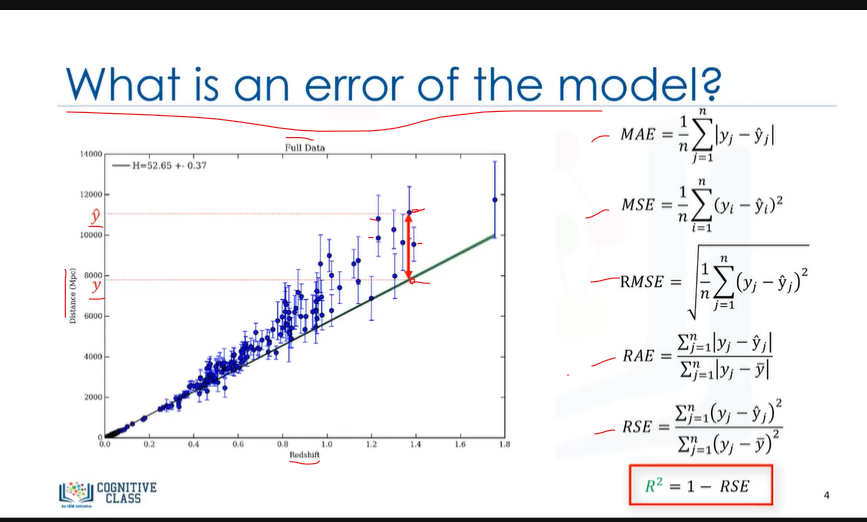
1. This is one of the most popular of the evaluation metrics because Root Mean Squared Error is interpretable in the same units as the response vector (or ‘y’ units) making it easy to relate its information.

**4)Relative Absolute Error (RAE), also known as Residual sum of square,**

1. Where y-bar is a mean value of y, takes the total absolute error and normalizes it by dividing by the total absolute error of the simple predictor.

**5)Relative Squared Error (RSE) is very similar to “Relative absolute error “,**

1. But is widely adopted by the data science community, as it is used for calculating R-squared.
2. R-squared is not error, per se, but is a popular metric for the accuracy of your model.
3. It represents how close the data values are to the fitted regression line.
4. The higher the R-squared, the better the model fits your data.
5. Each of these metrics can be used for quantifying of your prediction.
6. The choice of metric completely depends on the type of model, your data type, and domain of knowledge.



1. Unfortunately, further review is out of scope of this course.
2. Thanks for watching!

**NEXT Video:**

### **Non-Linear Regression (7:35)**

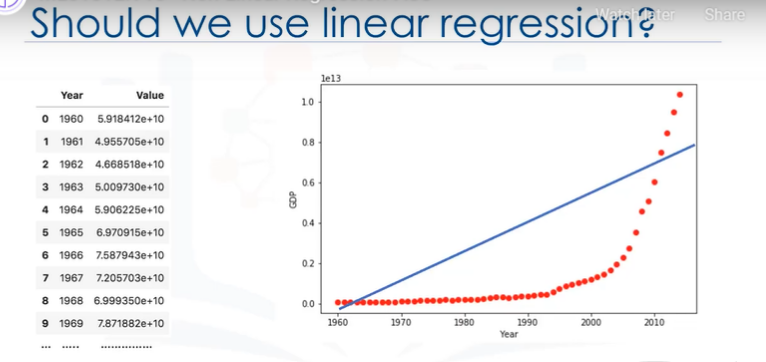


we’ll be covering non-linear regression basics.

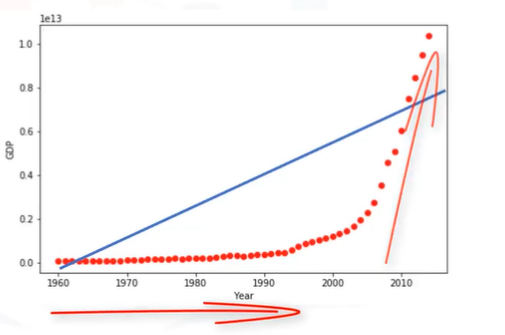
* These data points correspond to China's Gross Domestic Product (or GDP) from 1960 to 2014.

1. The first column, is the years, and the second, is China's corresponding annual gross domestic income in US dollars for that year.

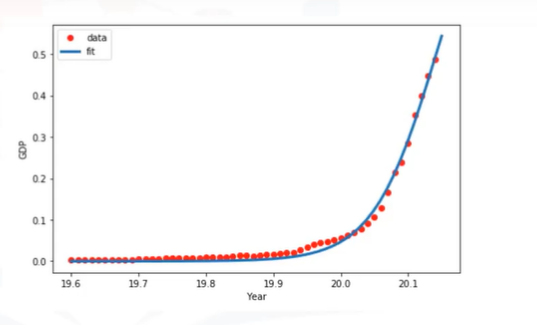
This is what the data points look like.

Now, we have a couple of interesting questions.

1. First, “Can GDP be predicted based on time?”
2. And second, “Can we use a simple linear regression to model it?”
3. Indeed, if the data shows a curvy trend, then linear regression will not produce very accurate results when compared to a non-linear regression -- simply because, as the name implies, linear **regression presumes that the data is linear.**
4. The scatterplot shows that there seems to be a strong relationship between GDP and time,
5. But the relationship is not linear.
6. **As you can see, the growth starts off slowly, then from 2005 onward, the growth is very significant.**



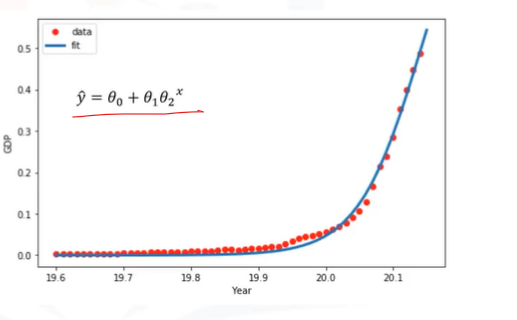
1. And finally, it decelerates slightly in the 2010s.
2. **It kind of looks like either a logistical or exponential function.**



1. So, it requires a special estimation method of the non-linear regression procedure.
2. For example, if we assume that the model for these data points are exponential functions,

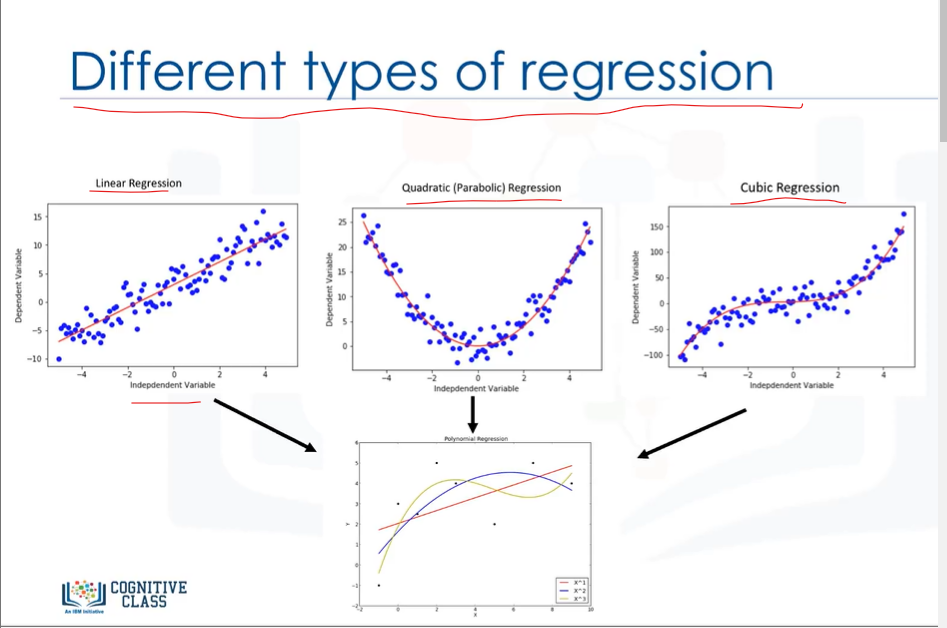
such as y ̂ = θ\_0 + θ\_1 〖θ\_2〗^x,

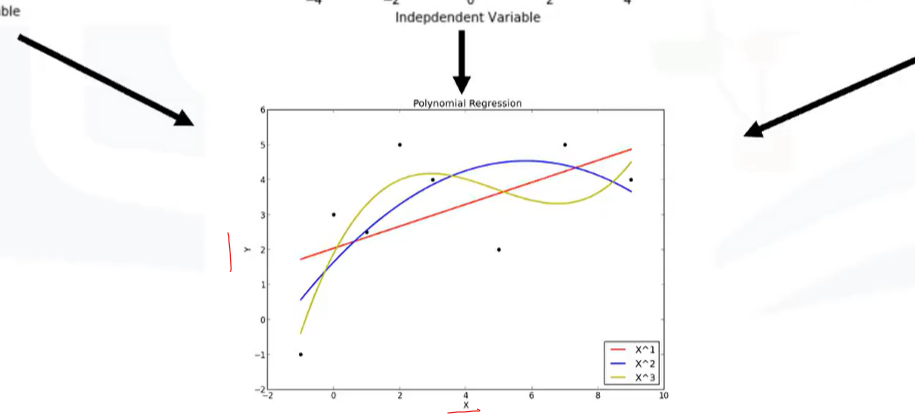
our job is to estimate the parameters of the model, i.e. θs, and use the fitted model to predict GDP for unknown or future cases.



**In fact, many different regressions exist that can be used to fit whatever the dataset looks like.**

1. You can see a quadratic and cubic regression lines here, and it can go on and on to infinite degrees.
2. In essence, we can call all of these "polynomial regression," where the relationship between the independent variable x and the dependent variable y is modelled as an nth degree polynomial in x.





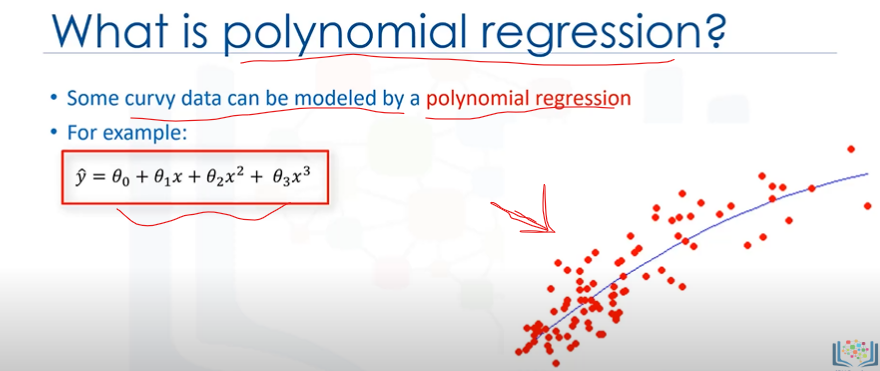
1. With many types of regression to choose from, there’s a good chance that one will fit your dataset well.
2. Remember, it’s important to pick a regression that fits the data the best.

**So, what is Polynomial Regression?**

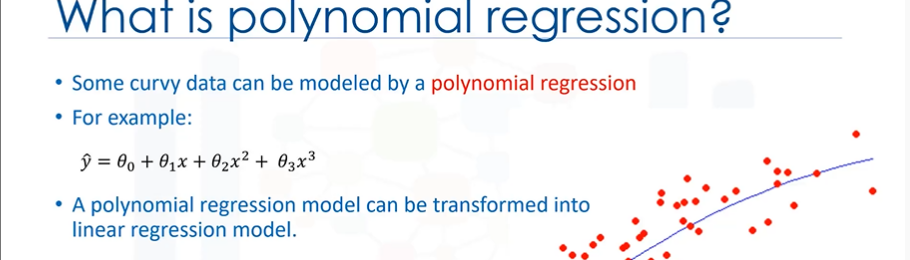
1. Polynomial regression fits a curved line to your data.
2. A simple example of polynomial, with degree 3, is shown as:

**y ̂ = θ0 + θ1x + θ2x² + θ3x^3**

where θs are parameters to be estimated that makes the model fit perfectly to the underlying data.



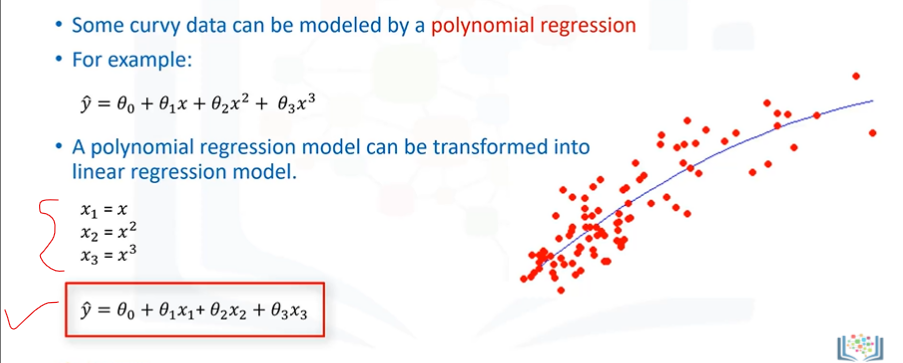
Though the relationship between x and y is non-linear here, and polynomial regression can fit them, a polynomial regression model can still be expressed as linear regression.



I know it's a bit confusing, but let’s look at an example.

1. Given the 3rd degree polynomial equation, by defining x1 = x and x2 = x^2 or x to the power of 3 and so on,
2. The model is converted to a simple linear regression with new variables, as:

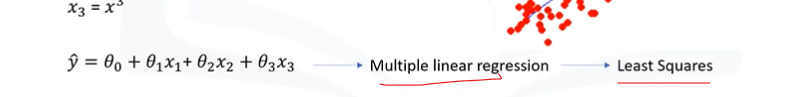
**y ̂ = θo+ θ1\*x1 + θ2\*x2 + θ3\*x3.**

****

This model is linear in the parameters to be estimated, right?

Therefore, this polynomial regression is considered to be a special case of traditional multiple linear regression.

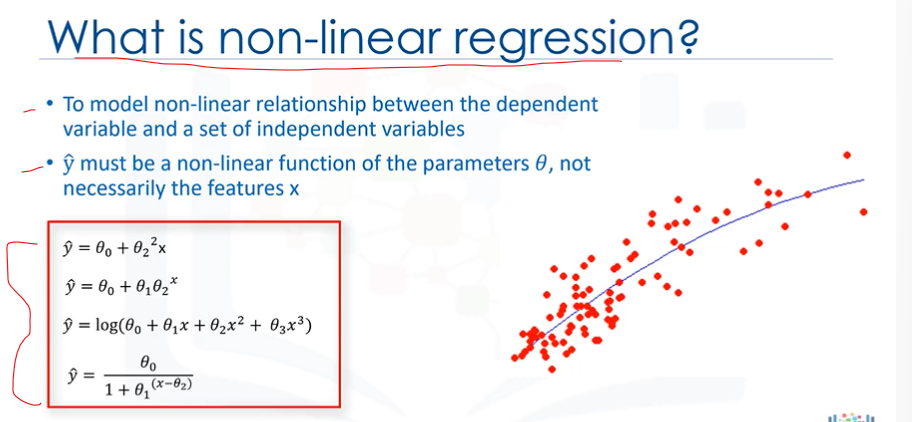
1. So, you can use the same mechanism as linear regression to solve such a problem.



1. Therefore, polynomial regression models CAN fit using the model of least squares.
2. Least squares is a method for estimating the unknown parameters in a linear regression model, by minimizing the sum of the squares of the differences between the observed dependent variable in the given dataset and those predicted by the linear function.

**So, What is “non-linear regression” exactly?**

1. First, non-linear regression is a method to model a non-linear relationship between the dependent variable and a set of independent variables.
2. Second, for a model to be considered non-linear, y ̂ must be a non-linear function of the parameters θ, not necessarily the features x.



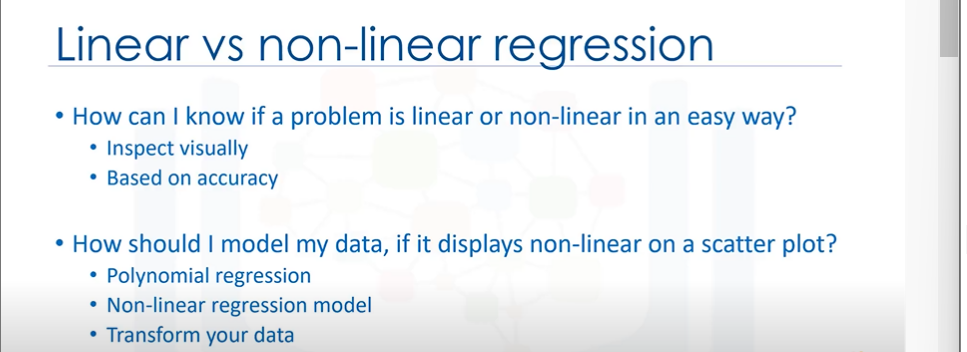
1. When it comes to non-linear equation, it can be the shape of exponential, logarithmic, and logistic, or many other types.
2. As you can see, in all of these equations, the change of y ̂ depends on changes in the parameters θ, not necessarily on x only.
3. That is, in non-linear regression, a model is non-linear by parameters.
4. In contrast to linear regression, we cannot use the ordinary "least squares" method to fit xthe data in non-linear regression, and in general, estimation of the parameters is not easy.

**Let me answer two important questions here:**

**First, “How can I know if a problem is linear or non-linear in an easy way?”**

* To answer this question, we have to do two things:

1. The first is to visually figure out if the relation is linear or non-linear.
2. It’s best to plot bivariate plots of output variables with each input variable.
3. Also, you can calculate the correlation coefficient between independent and dependent variables, and if for all variables it is 0.7 or higher there is a linear tendency, and, thus, it’s not appropriate to fit a non-linear regression.
4. The second thing we have to do is to use non-linear regression instead of linear regression when we cannot accurately model the relationship with linear parameters.



**The second important questions is, “How should I model my data, if it displays non-linear on a scatter plot?”**

1. Well, to address this, you have to use either a polynomial regression, use a non-linear regression model, or "transform" your data, which is not in scope for this course.

**QNA**

### **Review Question 1**

Train and Test on the Same Dataset might have a high training accuracy, but its out-of-sample accuracy can be low.

**True**

False

### **Review Question 2**

Which of the following matrices can be used to show the results of model accuracy evaluation or the model’s ability to correctly predict or separate the classes?

**Confusion matrix**

Evaluation matrix

Accuracy matrix

Error matrix

Identity matrix

### **Review Question 3**

When we should use Multiple Linear Regression?

**When we would like to identify the strength of the effect that the independent variables have on a dependent variable**.

When there are multiple dependent variables.