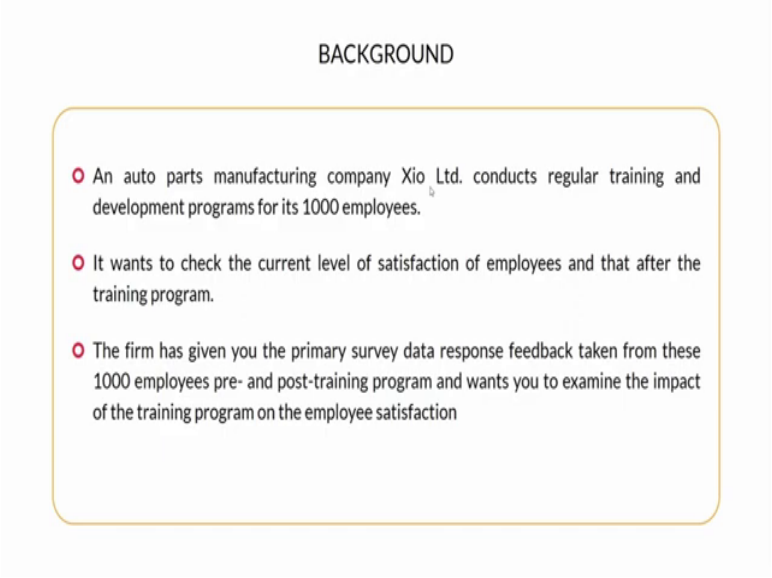
**Artificial Intelligence (AI) for Investments**
**Prof. Abhinava Tripathi**
**Department of Industrial & Management Engineering**
**Indian Institute of Technology – Kanpur**

**Lecture – 21**

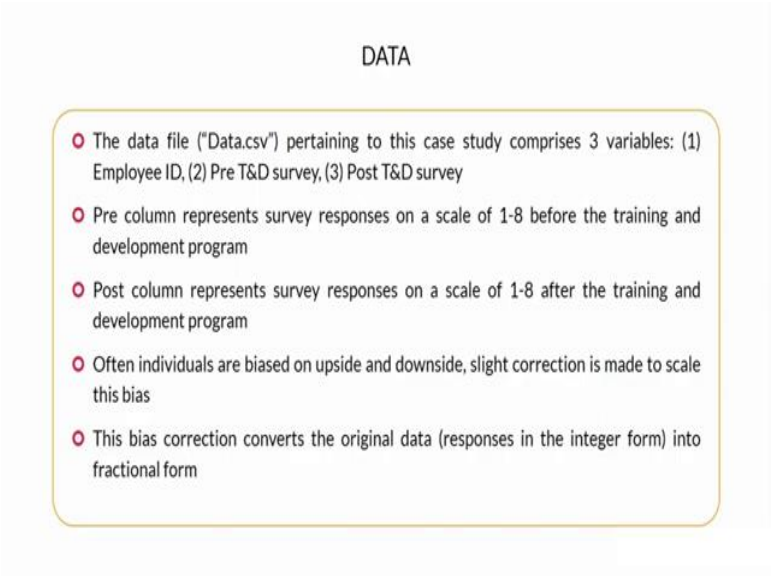
**(Refer Slide Time: 00:13)**




Let us start with this case study problem on auto parts manufacturing company Xio Limited. In this case study we will apply the concepts learned on the topics related to inferential statistics and hypothesis testing. An auto parts manufacturing company is Xio Limited, conducts regular training and development programs, for it is 1000 employees. It wants to check the current level of satisfaction of employees and that after the training program.

Here the chief data scientist in the consultant firm engaged by Xio Limited. The Firm has given you the primary server response data and the feedback taken from these 1000 employees pre and post training program and wants you to examine the impact of the training program on the employee satisfaction.

**(Refer Slide Time: 01:00)**



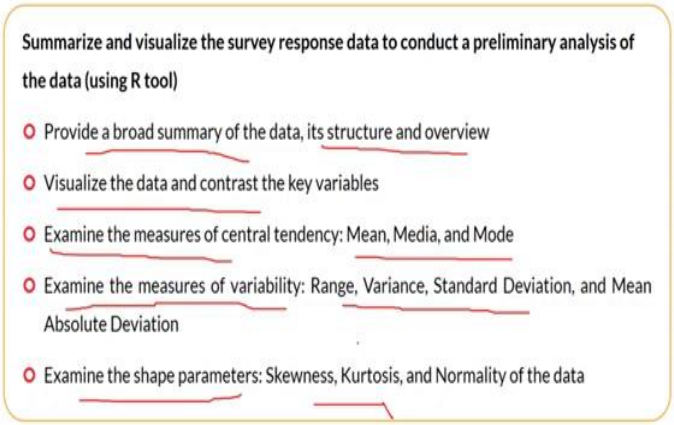

The data provided which is in the data file data.csv to you comprises three key variables. First variable is employee ID, second employee response data. Before conducting the survey, this data is taken on scale of 1 to 8. The data file uses the variable name pre to reflect this data. Third, we have employee response data after the survey, this data is taken on a scale of 1 to 8. The data file uses the variable name post to reflect this data.

Both of these variables are extremely important from the analysis perspective. Very often, individuals are slightly biased in their responses. Some are biased on the upside, while others are biased on the downside. Based on the response of these individuals to survey questions, the responses are corrected for this bias. This bias is corrected through scaling the data and that is why the final data is in the form of fractions.

Unlike the original data which was collected in integer form on a scale of 1 to 8, we will analyse this data using our programming language.

**(Refer Slide Time: 02:11)**

Case Analysis




Let us start the case analysis by summarizing and visualizing the survey response data to conduct the primary analysis using R tool. As a first step towards the data analysis, we will understand different ways of reading data in R. Next, we will summarize the data in R. There are various commands and functions that help us in getting a brief overview, summary and structure of the data.

There are various ways in which summary measures help us got broad trends available in data. Using these summary images, we will try to understand the key differences and contrast the pre and post training survey responses. Next, we will plot data using plot functionalities of R, we will also understand how to improve the aesthetic aspects of these plots. Subsequently, using plot technique, we will superimpose the survey response, pre and post training data.

We will also examine the density distribution of these variables to understand the impact of training. Next, we will examine the measures of central tendency for our pre and post variables. The measures of central tendency examined here are mean, median and mode. In addition, we will also analyse the quantile wise distribution of data. Overall, these statistics will help us gauge whether there is some significant impact of training on implied satisfaction.

Next, we will examine the measures of variability. The measures of variability studied here include range, variance, standard deviation, mean absolute deviation that is mad. These measures will help us understand the variability in the variables pre and post training on survey responses. Very often, distributions are simply defined by two parameters, including mean and variance.

And therefore, the variability of key parameters in the study is very important to understand. Lastly, we also examine the shape parameters. This includes skewness which is a measure of symmetry kurtosis which is the measure of weakness and normality of data. We will first compute these measures for a simulated, normal distribution and then compare these measures for our variables, pre and post.

That is survey responses before and after training did. Moreover, we will also use our functionalities to statistically test whether these deviations from normality are indeed significant or not. This includes test of normality like Jarque-Bera test, test of skewness and test of kurtosis. Lastly, we will also plot the histogram of the data to visually inspect and confirm the deviations from normality.

**(Video Starts: 04:51)** Now that we have understood the data, let us start by reading and writing it. First, we will try to analyse the data and as a best practice, I would recommend that we set our working directory first. So, we will click on the session button. We will click on this set working directory and then choose the directory appropriately. Once you click on this open, you will notice that a command has appeared on the console window.

As the best practice I would recommend that you copy paste this command on your current script. The idea here is that in future, whenever you are working on this script later, you need not remember the exact location where your current working directory and all your data is available. Notice on the file tab here you will find all your current working directories and data available directly there.

If you want to read a data file, let us say data.csv, you can click on it. Once you click on this, you will get an import data set button and if you click on this import data set button, it will give you the shape and form of data, how it is looking? If you click on import, this data will be downloaded and uploaded in your current book directory. In the current form, there are some options for example, you can choose first row as names trim, spacer and so on, as may be found suitable.

And you click on import and you will again notice first data is visualized in short form on your current tab. And also you can notice that a command has run which you can directly use in

future as well. That is one way to read the data here. Also, the data variable is appearing on your current working environment. That is also there. If you click on it again, the data will be visualized briefly on your current script.

Another way to import data is to click on this import data set appearing in the environment, tab and then text button. If you click on this text button, it will give you some options to read the current work in directory and you can click on the data file. Double click, and it will again give you some form of visualization of data, how it look like if it is imported into the current working directory.

If you click on it again, command file is run. You can see that on my console window, the command file that has run. Lastly, there is another way to read this data is through read.csv command. For example, I can write this command notice I have not given the exact location of the data, from where it is to be read. Simple reason is I have already set my current work directory.

It is one of the major benefits of setting the current work directly that you need not specify every time whenever you are reading or writing the data it will automatically take this particular location. However, if you want to read and write from some other place then you need to specify that location separately. But if you are reading the data from the current working directory or writing it there then you need not specify the address location.

Now that we have read the data, let us start by examining the nature of this variable. As a first step, you would like to know the nature of this data and we will examine the nature of this data by running this class command notice that this data is a data frame variable a very important class of variable in our format. A data frame is a kind of variable that is a collection of different columns, each column itself is a variable and each column variable can be of different nature.

For example, it can be numeric; it can be character; it can be factor and so on. Let us have a brief overview of this data now. So, I will use the head command and once I run the head command on data, I can see some of the elements initial elements appearing on my console tab. I can see those elements. There are three variables it seems, as we have discussed earlier about this data first is employee ID.

Then pre which is the survey response variable before training and development then post which is the survey response variable post training and development. We can also see some of the last elements by running this tail command and you can see some of the last elements starting from 995 to 1000 have appeared. If you want to check the dimension of this data, we can simply run dim command and you can see there are 1000 rows.

Each row is an element, as we already know from this data frame and there are three columns which is also appearing on the console foreign. Now, let us summarize this data and we will use the summary command. It will be interesting to note that all the three variables have been summarized the pre and post both of the variables have been summarized like they are numeric variables.

You can see their minimum first quartile median, mean, max, etcetera are appearing. Also, you can clearly see, as per this summary, there is a shift from pre to post. There is a certain shift that is appearing. It gives you a look and feel that there is a certain shift after training but is still more analysis more statistical procedures are required to establish that to a good extent.

One problem, as you would notice that employee ID variable has also been summarized like a numeric variable. Of course it is not a numeric variable but the way R works it tries to summarize each variable in a manner that there is a minimum loss of information. So, it has summarized it in a numeric form. Next, another way to summarize the data or to see what are the contents of this data frame are to use structure, str command.

And as I run this str command notice that it tells us that there are three variables we can clearly see on console window. It gives me a brief overview of this data frame. It tells me that there are 1000 observations and three variables that is first set of information. Then it also tells me that there are three variables, employee EMP, pre, post. And what is the nature of those variables? For example, employee, EMPID is taken as integer.

Although it is not exactly an integer variable but R is recognizing it as that we can also change the way, I recognize a certain variables. Then we have pre which is the survey response variable before training R has recognized that as a numeric variable, it is a fraction of course. Again this post variable which is again a numeric variable in a fractional form, remember from our earlier discussions that the data has been scaled from integer to numeric for bias correction.

It also tells us that in order to access these variables inside the data frame; we need to use dollar sign. For example, if I want to access the pre-variable, I need to type data and then dollar sign. This is called tab completion, a very important property of our data. So, with this tab completion I can click on tab button after dollar sign and you can see all the three columns are appearing I can select one of them.

For example, I can select preview. So, this is how we can summarize the data see the structure of data frame in which all the variables are there. And we can also get a brief summary and brief overview of our data that is available for analysis. With this now that we have understood the summarizing the data and we have now the overview of data, we will move to the visualization part and plotting of the data.

```
#code
setwd("....")
Data=read.csv("Data.csv")

class(Data)
head(Data)
tail(Data)
dim(Data)

summary(Data)
str(Data)
Data$Pre
```

Now that we have got a brief overview of data, it is structure and summary. Let us start by visualizing the data. In this video we will see how to visualize the data using R? A lot of interesting insights can be obtained by simply visualizing the data. For example, you can see if there are any outliers or with the help of density plot. You can also see if data follows a particular distribution like normal distribution.

A very basic and simple plot command available with our, for example, plot command will apply on a data variable. Notice, our plot tab I will zoom it to improve the visualization notice, both the variables pre and post are appearing here. For pre-variable, the range is 3 to 5 which indicates it is observations. And similarly the value range for post is 4 to 6. This clearly suggests that because of the training, there is a clear increase in the employee satisfaction pre to post.

This provides us with some initial intuition to the impact of training and development program. We can also use this plot command for a single variable pre and the plot appears on my plot window. I can either save it as image. For example, I can save it as png or jpeg image. I can change the width and height of this plot. Also, you can export this as pdf or you can copy it to clipboard as well and subsequently paste it on your current working directly, maybe a word or ppt document.

Then you can do a lot more to improve. It is aesthetics, for example, as a starting point let us plot this pre-variable and in the argument type I am specifying p. This will tell R that the plot needs to be made in point form which I have already done, so, it will not make a difference. However, I could have instead of p used l as well which would have plotted in the line form but, as we can see, the line plot may not be too appropriate for this kind of data.

So, we will stick to the point plot we can also add some colours here, for example, I can specify the colour with col argument as red. You can see the points are plotted now in red colour. We can further improve this plot by adding legends and access labels. For example, we can add the main central heading; maybe I can name it as graph. I can also add axis labels, for example, x axis label by xlab argument set to data.

```
#code
plot(Data)
plot(Data$Pre)

plot(Data$Pre, type="p")
plot(Data$Pre, type="p", col="red")
plot(Data$Pre, type="p", col="red", main="Graph", xlab="Data", ylab="Frequency")

plot(density(Data$Pre), main="Density-Graph", xlab="Data", col="red", lwd=3, xlim=c(2,7))
lines(density(Data$Post), col="blue", lwd=3)

legend("topleft", c("Post","Pre"), fill=c("blue", "red"))
legend("topright", c("Post","Pre"), fill=c("blue", "red"))
legend("bottom", c("Post","Pre"), fill=c("blue", "red"))
```

Similarly, I can set the colour to red which is already there, so, we will run this command and notice once I run this command central heading appears and I can also add y-axis label. Maybe I can put it as frequency, so, it will change the y-axis label and so on, lot of aesthetic improvement we can do here to make it look more professional. Now, many times we are interested in density plots a simple command or function which is density function I can use here and apply it on my pre-data to plot density.

Now here in the central argument main heading, I can put it as density graph. I can also specify my x-axis label by xlab argument as data. I can add a colour of red I can add certain more arguments, maybe I can add a line width let us say a line bit of one. Let me run this command as you can see. Now, I have obtained the same plot that I wanted in red colour, the density plot.

Probably the line width is slightly low, so, I can increase the line width by using lwd command, setting it to 3. So, now my line width has improved. Now, I would like to superimpose the post observations after the training and development program. For that I need not write the plot command again. I need to use the same plot because I want to superimpose and in that I will use the lines command.

And I use the density function on my post data to tell R that the line corresponding to density plot on post variable are to be added superimposed on the original plot. I give it a colour of blue to differentiate between the old and new plot and I get a line width of 3. If I run this, you can see that now both the graphs are appearing superimposed on one plot but there is some problem with this because x axis width is limited up to 5.

However, it seems the post variable because there is a shift in the variable is not completely coming inside the graph. So, I will make some small modifications by adding a limit, excellent argument which specifies the limit of x axis labels and I will specify the limits between 2 to 7. So, on the right side, I am increasing the graph width a little bit. And you can see now when I am plotting the plot original plot there is certain additional space on the right side which will give me some allowance.

And now I will learn the lines function. Where post variable is there and you can see now, both of these plots are appearing within my plot boundaries. And clearly they are showing that there is some increase in the post variable after training and development. Now, I can also add

legends here. Very simple I will use the legends function and specify the location as maybe top left.

Then I need to specify the names I will go with the same convention as post and pre. And then I will add the fill colour I will again follow the same convention as blue for post and red for pre. Notice that legend has appeared I can change the location of this legend very easily. For example, instead of top left, I can give it top right or for that matter, on the bottom side also, I can give it and so on.

So, a lot of permutation combination as possible. However, this graph clearly indicates that there is a certain impact of training and development pre and post on the employee satisfaction. Now that we have understood the data visualization, we will move to examine the measures of central tendency with the R implementation that gives us more robust and quantitative estimates.

Now that we have understood the data visualization, we will move to examine measures of central tendency that provide us a more robust estimate. A measure of central tendency is a summary statistic that represents the centre point or typical value of a data set in statistic. The three most common measures of central tendency are the mean, median and mode. We will start with mean and it is a sum of observations divided by the total number of observations.

It is also defined as an average which is some divided by count. In R the syntax mean is very simply employed like this. Mean inside I am putting my variable of interest which is pre and I can get the value by running this command which is 4.007. Now, instead of printing this value, I can also assign it to a new variable which is m1 by running this kind of command this is assignment operation. Now, the variable m1 contains this value of mean for pre-variable.

Similarly I can use almost similar command to generate the mean for my variable post which is the survey responses after the training program. It is quite easy I can print the mean of post which is 4.998. I can also assign it to a variable m2. Notice that this variable m2 is 4.988 which is more than the mean of pre which was 4.007 which is an indication that the mean has increased very often mean also represents the expected value of a variable.

For example, when you are computing expectations, often you do not have probabilities assigned to you. So, you used historical observations, assuming that each observation has a equal rate and therefore, the averages would represent equal probability weighted values. And therefore, also represents the expectations of the variables and that is why mean is also very important. Sometimes the variables contain any observation which is not available.

In those cases if you hunch or your data collection exercises, suggest that there may be some na kind of variables in your data then you use na.rm = T true. That means you are telling your R console that this variable may have na observations and if there are kindly remove them, if you run that now your mean will be computed, ignoring all the new observations. So, this is how you tackle na observations in the data foreign.

```
#code (Mean)
mean(Data$Pre)
m1=mean(Data$Pre)
m1

mean(Data$Post)
m2=mean(Data$Post)
m2

m1=mean(Data$Pre, na.rm = T)
```

Next, we will compute median, median is also very important measure of central tendency. It is the middle value of data set. That means it divides your data set into two halves. So, first, we computed mean now we are computing median. The syntax to compute median is also very simple. We will type this median command and first we will try to see the median of our variable free. The median of this variable is 3.99.

```
#code (Median)
median(Data$Pre)
m1=median(Data$Pre)
m1

median(Data$Post)
m2=median(Data$Post)
m2
```

We will also assign the value to a variable called m1 we can do that very easily like this. Similarly, I can compute the median of my post variable. Notice the median is 4.98. I can again

assign this value of median to my new variable m2. The value of this variable m2 notice is 4.98. Now, again, an indication that m2 is greater than m1 which suggests that there is some impact of training on the median of the data.

After median, we will compute another important measure which is mode. For mode this is the value that has the highest frequency in a given data set. Also, if you are talking in terms of probability density mode, is the variable where highest probability density is there for given set of values. Now, one way to run this is install a package called static. So, we will use this installed dot packages command to install this new package called static.

Once you install this package started what you need to do is? You need to simply add this to your current working library by running this library startip command. Once you install this library as static, you can very well, use the function, mfv to compute probability density. The way this function works, it organizes or ranks observations according to their frequency. That means those observations which are highest in terms of frequency or they have the highest probability density are put earlier than those that have lower probability density.

```
#code  (Mode)
install.packages("statip"
library(statip)

head(mfv(Data$Pre),5)
head(mfv(Data$Post),5)
```

So, in case, if you want to see the top 5, 10 or some initial values you run head command. As we are very familiar with that command, we can print a set of initial values. For example, let us say I want to print initial five values. I can run head commands like this and it will print my initial five values. So, you can see these are some of my initial five values printed on my console for my pre-variable.

Similarly, I can run this for my post variable. Notice that the mode of my pre-variable versus post variable is not same. In fact, it appears that there is a considerable increase in the mode of the data after the training and development program. So, in this fashion, like median and mean, mode, is also helping us understand the impact of training on our satisfaction of employees.

Lastly, we will also examine the quantile measure, a very important and interesting measure. What are quantize again? Quantizum are a measure that divide data into different, different buckets. For example, if you are dividing your data into four buckets of equal sizes, you will call it qualities. That means Q1, Q2, Q3, Q4 and each bucket would involve 25 percent of the observations.

Similarly, if you divide your observations into 10 equal buckets, you will call them deciles. That means each bucket would know 10 percent of the data sets is the smallest division if you want divide them into 1 percentile, 1 percentile bucket then you call them quantile. Now, it is quite easy to compute quantile in R with functional quantile use function quantile. Let us see the pre-variable quantile by default are computes quantiles for you, starting from 0 percentile to 100 percentile.

It divides data into four buckets and you get the corresponding observations. In this particular case, you get 0 percentile at 3, 25 percentile highlight 3.53, 50 percentile at 3.99 and so on. Similarly, I can compute the quantize for variable post, as you can see for each of the quantiles 0, 25, 50, 75. I get a relatively higher value as compared to variable pre which again conforms to our earlier intuition that there is definitely some increase in the custom in the employee satisfaction.

```
#code (Quantiles)
quantile(Data$Pre)
quantile(Data$Post)
```

Now, let us say instead of these default quantiles, you want to compute certain specific quantities. Let us say you want to compute 95 percentile for your pre-variable. Quite easily computed you can simply type this syntax. You can specify the quantile that you want to compute and it will tell you that 95 percentile is 4.991 for pre-variable. Similarly, you can compute 95 percentile for your post variable.

And it appears to be 5.897 which clearly indicate that there is some increase in the satisfaction. Now, you need not restrict yourself to only a particular set of quantile and you can use a very important function called sequence function to generate quantiles, as you may want. For example, I can use a sequence function like this and specify the sequence starting from 0 ending at 1 with an interval of 0.2.

*#code (Quantiles)*
*quantile(Data$Pre, 0.95)*
*quantile(Data$Post, 0.95)*

So, it will generate a new sequence of quantile which will cut off at points 0 percentile, 20 percentile and so on. Let us see the output, as you can see a very interesting set of observations are appearing for five buckets that cut at 20 percentile, 40, 60, 80 and 100. Exact same sequencing we can produce for post variable and it will give us again the five divisions but you can clearly see that observations corresponding to each of the quantiles that is 20 percentile, 40 percentile, 60 percentile.

*#code (Quantiles)*
*quantile(Data$Pre, seq(0,1,0.2))*
*quantile(Data$Post, seq(0,1,0.2))*

Have increased considerably as compared to their pre-training values. Overall, these observations clearly suggest that now there is a shift in the distribution of our variable of interest which is employee, satisfaction after the training. So, with this, we understand that measures of central tendency clearly indicate that there is some shift in the distribution of our employee satisfaction.

In the previous video, we examined the measures of central tendency in this video we will discuss about the measures of variability and how to estimate them? So, we will talk about measures of variability in this video, measures of central tendency yielded information about the central or middle part of the data set. However, business researchers can use another group of analytical tools to measure the measure of variability to describe the spread or that is dispersion of data set.

Using measures of variability in conjunction with measures of central tendency, makes it possible for a more complete numerical description of data. There are a number of measures of variability. Some of the most important ones include range variance standard, deviation and mean absolute deviation that is mad. So, we will start with the range measure of variability, the range describes the difference between the largest and smallest data point in our data set.

That means the bigger the range the mode is the split of data and vice versa. The syntax to measure the range is quite easy range and then we will apply a data set which is pre. We can see the range of pre data is 3 to 4.99. I can also check the range of my post data which is again quite easy it is 4 to 5.99. The results clearly suggest that post training exercise the range has shifted.

```
#code
range(Data$Pre)
range(Data$Post)
```

Although the spread of the range is not very different but there is a shift in the range that is shift in smaller and larger values. Our next measure of variability is standard, deviation and variance. So, first we will talk about variance. Variance measure is the average of squared deviations about the arithmetic mean for a set of numbers. The syntax of this variation measure is var which is short for variance.

```
#code
var(Data$Pre)
var(Data$Post)
```

And then we will put our data which is pre-variable. The variance is 0.32 very similarly I can compute the variance of post data very close to earlier value which is 0.325 which we also got some inkling from our range measure that there is not much change in the variation only there is a shift in data. The third measure of variability is standard deviation. Standard deviation measure is square root of variance that is standard deviation measures the dispersion of data set relative to it is mean.

It is defined as the square root of variance and if the data points are further from the mean there is a higher deviation within the data set thus the more they spread outside the data the higher the standard deviation. The syntax to compute standard deviation is quite simple, sd data dollar tree which is our pre-variable, the standard deviation which is 0.57. Similarly, I can compute the standard deviation of my post data which is very similar to the earlier value.

```
#code
sd(Data$Pre)
sd(Data$Post)
```

Again 0.57 which again confirms that the spread is almost similar. The last measure that we are discussing is mean absolute deviation. Mean absolute deviation often referred to as mad measure. This measure is the average of absolute values of deviations around the mean for given set of observations. The computation and syntax is quite simple mad data dollar tree. So, the mean absolute deviation is 0.721.

```
#code
mad(Data$Pre)
mad(Data$Post)
```

Similarly, I can compute the mean absolute deviation for data post. Again, not very different, slightly lower but 0.70, so, not much shift in variation, pre and post training program. However, as we have seen the range and other measures of central tendency, we can clearly say that while there is not much difference in the dispersion of the data before training and after training but clearly there is a shift in the distribution.

That means the lower, as well as higher values, have indicated a kind of shift on the positive side after training. However, to get some more idea about the variables and the impact of training activity, we will also examine the measures of shape of inferential statistics. Now, that we have made ourselves familiar with the measures of central tendency and the measures of variation.

In this video we will discuss about the measures of shape and their implementation with our software. We will talk about measures of shape in this video and in order to implement these measures, we need to install two very important packages. First is time series package, the second one is moments package. These two packages provide very important functionalities for implementation of some of the shape measures.

```
#code
install.packages("tseries")
install.packages("moments")
library(tseries)
library(moments)
```

Once you install these packages, you need to put them in your current working library with the library command like this. First I will run my time series package and then I will run library

command for my moments package. Now, once you run these libraries, they will be installed in your current working directory. First and foremost, we will talk about the measure of skewness.

The distribution of data in which right half is mirror image of left half is said to be symmetrical. One example of this symmetrical distribution is normal distribution or bell curve. Skewness is obtained when distribution is asymmetrical or lack symmetry, as you would have guessed, normal distribution has 0 skewness, so, in order to check the skewness of digital distributions.

First, let us start with a normal distribution, let us generate a normal distribution and give it a name Norm. We will use our simple our Norm command to generate 10000 observations for normal distribution and we will keep using this for our investigation of shape measure. So, first, let us compute the skewness of this non-variable. As you can see, it is very close to 0 which suggests that the distribution has a status which is closer to 0.

Let us compute the skewness of R pre-variable. It is also quite close to 0 which suggests the data has a skewness of 0. Let us compute the skewness of our post variable. As you can see this is also very close to 0. That means our pre and post variable conforms to 0 skewness and which is quite similar to normal distribution. However, this is just a measure and it does not give any statistical confidence or assurance.

So, we can also perform agostino test which is a more of a statistical measure of skewness. So, this agostino test, let us first start with our Norm variable. As you can see the p-value is 0.34 which suggests that the null hypothesis cannot be rejected. That means the data does not have a skewness the same agostino test we can perform on our pre-data. You can see again the hypothesis cannot be rejected.

```
#code
Norm=rnorm(1000)

skewness(Norm)
skewness(Data$Pre)
skewness(Data$Post)

agnostino.test(Norm)
agnostino.test(Data$Pre)
agnostino.test(Data$Post)
```

That means the data does not have a skewness. Same test we can perform or post it also. Again, the hypothesis cannot be rejected which suggests that our data does not have skewness whether it is pre or post. Next, we will talk about kurtosis, so, kurtosis is a measure of weakness and we will discuss more about this kurtosis measure. Kurtosis describes the amount of weakness of a distribution.

Distributions that are high and thinner referred to as leptokurtic. For them the kurtosis value is greater than 3. Similarly, the distributions that are flat and spread out are referred to as platokurtic distributions for this, the value should be less than 3. And as you would have guessed, between these two types of distribution, there are normal distributions which are also called mesokurtic distributions. For them, the value should be equal to 3.

So, let us try and compute this kurtosis value for our normal distribution first kurtosis very simple to compute. Let us use this for a normal distribution and it seems the kurtosis is very close to 3 which was expected because we simulated a normal distribution. Let us compute this kurtosis for our pre-data. You can see the kurtosis is not close to 3 it is in fact much far away than 3. Similarly, I can compute kurtosis for my post data also.

Again, the kurtosis is further away from so, it indicates that probably these pre and post variables do not have kurtosis that resembles a normal distribution but to perform a statistical test which will give us more confidence and more statistical significance. We will use anscombe test the test is divided moments package first, we will perform it for the normal data. And as you can see, the hypothesis that its kurtosis is equal to 3 cannot be rejected.

That means kurtosis indeed equal to 3 for that normal distribution will perform the same test for our pre-data. You can see the kurtosis is clearly not equal to 3 the hypothesis is rejected very significantly, so, the kurtosis is not equal to 3. Let us again test for the post data again. The cutoff is not equal to 3 which means that for this distribution, the cutoff is not equal to 3.

```
#code
kurtosis(Norm)
kurtosis(Data$Pre)
kurtsosis(Data$Post)

anscombe.test(Norm)
anscombe.test(Dtata$Pre)
```

*anscombe.test(Data$Post)*

Now we will test for an overall hypothesis the data is normal or not. So, we will test for the hypothesis of normality. In statistics jarque bera test is widely available to test the normality of the data. Test is available in time series package as well as moments package. So, first we will implement this jarque bera test for time series package which uses jarque bera.test command.

*#code*
*jarque.bera.test(Norm)*
*jarque.bera.test(Data$Pre)*
*jarque.bera.test(Data$Post)*

And we will use it for our normal variable which is norm and you can clearly see. We cannot reject the hypothesis that it is normal and that means this. Obviously we assimilated the normal data itself, so, we cannot reject the hypothesis. Let us test this for our pre variable now, as you can see for pre-variable, we are able to reject the hypothesis with lot of confidence indicating that the data probably is not normal.

For the post variable if we test this hypothesis again, we are able to reject this with a lot of confidence. Let us try to run this jarque bera test with the different package. So, we will run this for our moments package which is the command, is the jarque.test and first we will run it with our normal variable. For the norm variable again obviously it was a normal distribution simulated, so, it cannot be rejected, the hypothesis cannot be rejected.

*#code*
*jarque.test(Norm)*
*jarque.test(Data$Pre)*
*jarque.test(Data$Post)*

Similarly, we will test it for our pre-variable and again, as we obtained the same result earlier, we have to reject the null hypothesis. That means it is not a normal distribution in a very similar manner. We can run our post data also which also suggests that it is not a normal data. Now, that we have run normal distribution test. Let us visualize this data again to see if indeed the visual inspection suggests some kind of presence of non-normality.

And for that very easily we will first start with the normal histogram. We all know what is histogram so, first, let us run with this hist command and I will perform this hist command on our normal data, to see how it looks. And you can see it is a very nice shaped bell shaped normal curve currently on bikes is you have frequency dimension. If you want instead of frequency, one can use probability I can put this feature probability equal to 2.

```
#code
hist(Norm)
hist(Norm, probability = T)
lines(density(Norm), lwd=3, col='red')

hist(Data$Pre)
hist(Data$Pre, probability = T)
lines(density(Data$Pre), lwd=3, col='red')

hist(Data$Post)
hist(Data$Post, probability = T)
lines(density(Data$Post), lwd=3, col='red')
```

And instead of frequency now you will have probability density. The diagram will remain simpler. Moreover, if you want to superimpose the lines of density over this histogram, you can very simply use lines command and you can plot density. Now, to make it more aesthetic appeal. You can add some further features like line width equal to 3 so, there is a solid line will appear.

And let us give it a colour of red so, now this histogram will give me a dotted solid red line which shows as we can see, on the graph, a normal distribution. And now let us plot the pre and post variables on the same block. So, let us start with plotting the pre-variable instead of norm we will use our pre-variable data dollar tree. Let us plot it is program. Obviously it is slightly different in fact than the normal distribution.

It appears to be symmetric but obviously the peak is not very similar which we would have anticipated, given our earlier stud estimations instead of frequency. Let us plot the probability density now, as you can see. Obviously, the diagram remains similar but we get an idea of probability density as well. We can also plot the lines, probability density lines and we can see the density lines up indicate that the kurtosis is very different from a normal distribution.

Again in a very similar manner I can compute the histogram and the density plots for post variable. Look at the histogram again, it is not in terms of ketosis, it is very different, it is symmetric, although around it is mean but it is still its kurtosis is very different than a normal distribution. We can again plot the density plot and plot the density lines. Also, so, now it is clear that this is not a normal distribution.

It is symmetric, though but since kurtosis is different. And why it is not normal? As we already saw the overall nomarlity hypothesis was rejected but it was not rejected on account of skewness but it was rejected on account of kurtosis. But an interesting question arises while we have been able to reject the hypothesis of normality for both of these data sets in subsequent applications of hypothesis testing, interval estimation etcetera will be making use of normal distribution, why?

So, as we have already studied in central limit theorem that distributions of sample means approach, normal distribution, when you have large number of samples, if sample size is more than 30. Using the central limit theory we will be able to use normal distribution to model these data sets. **(Video Ends: 48:57)**