INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
NPTEL
NPTEL ONLINE CERTIFICATION COURSE

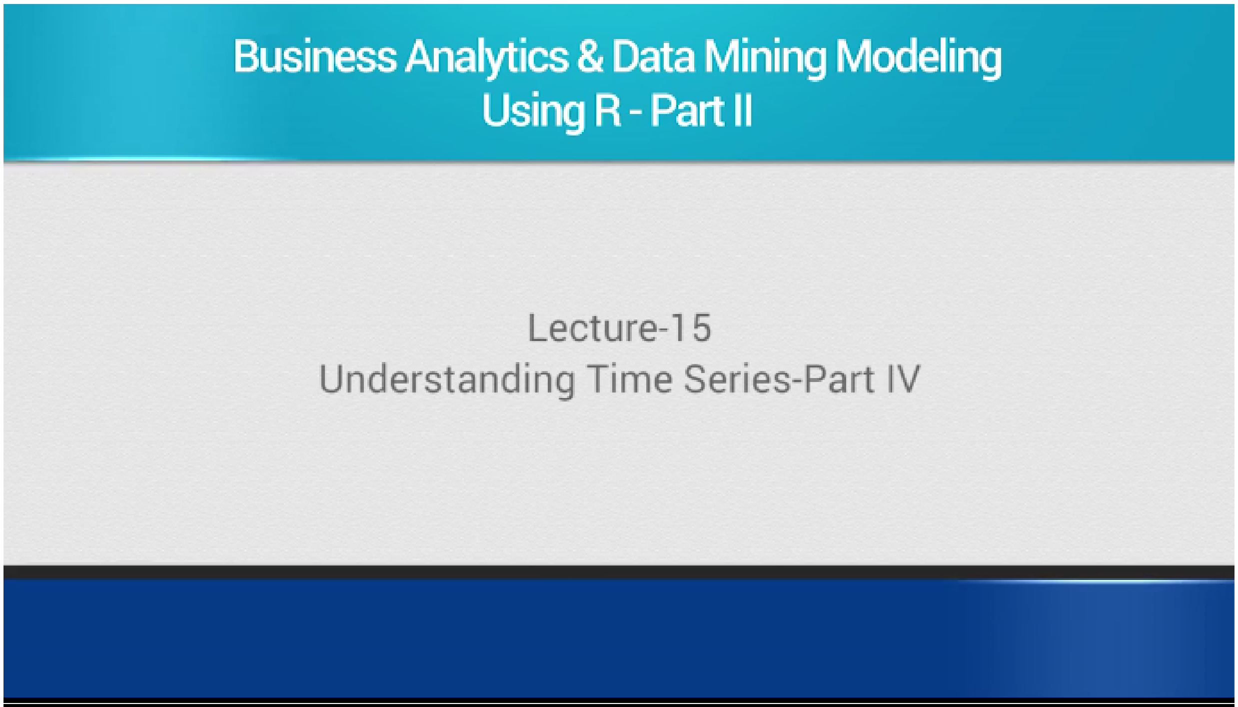
Business Analytics & Data Mining Modeling Using R – Part II

Lecture-15

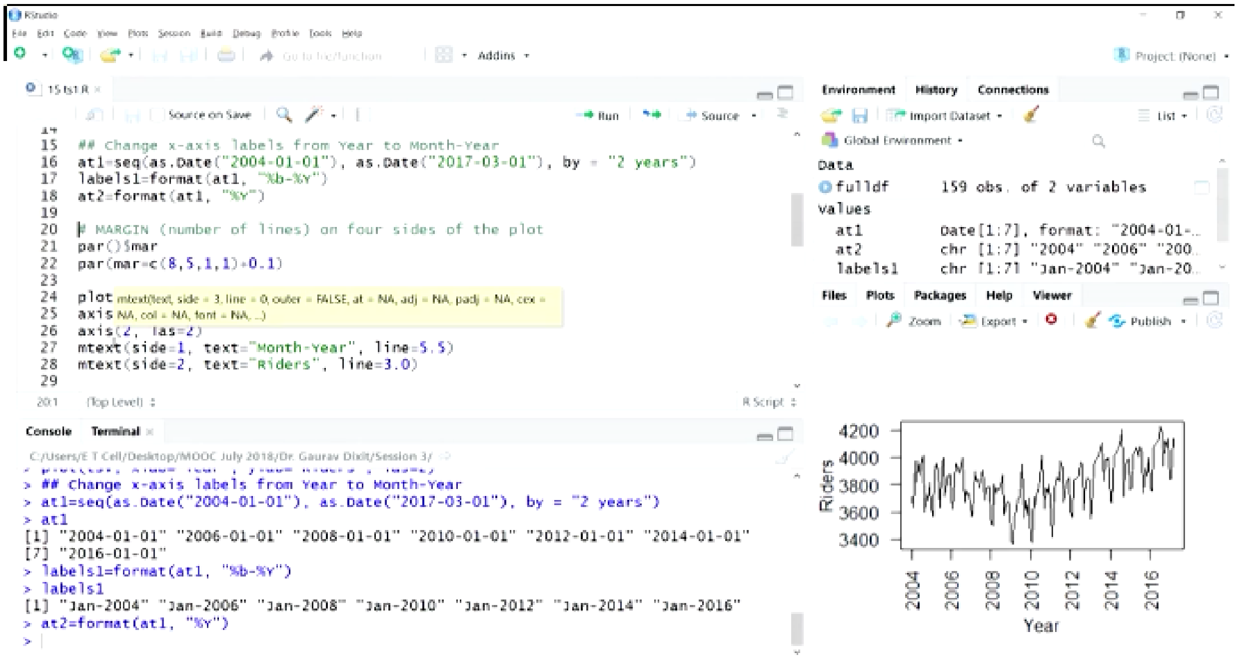
Understanding Time Series– Part IV

With

Dr. Gaurav Dixit
Department of Management Studies
Indian Institute of Technology Roorkee

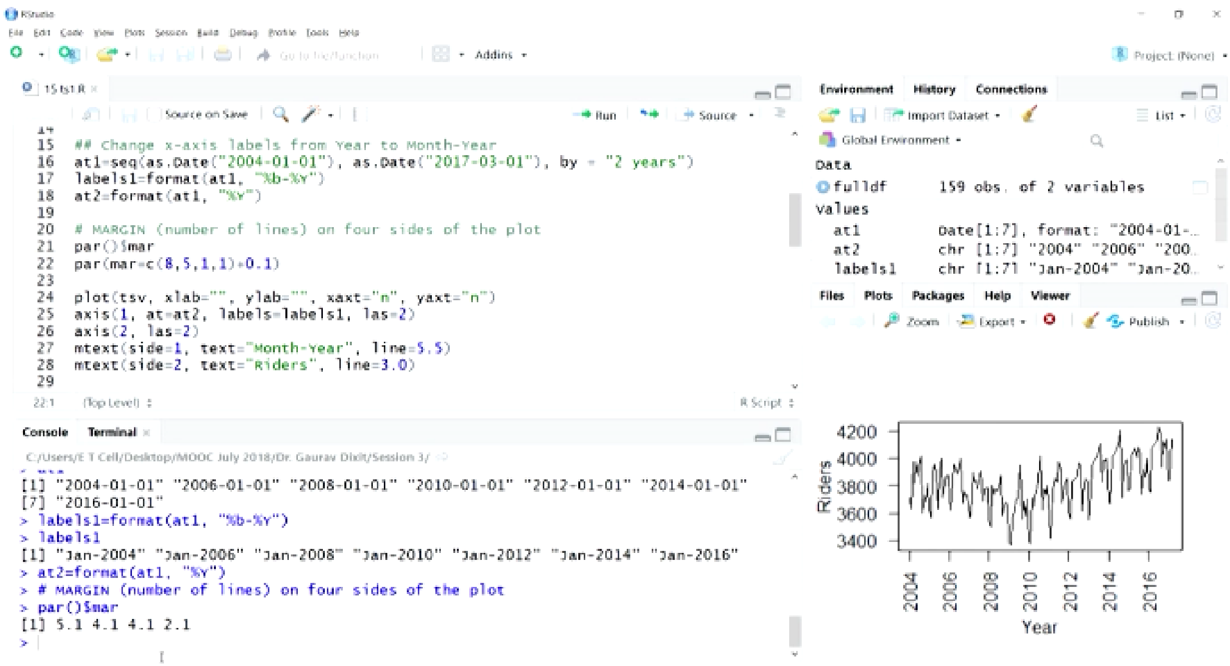


Welcome to the course Business Analytics and Data Mining Modeling Using R – Part 2, so in previous lecture we were discussing on the time series, specifically time series component, so we had completed our discussion on time series components, we talked about level, trends, seasonality, and noise.



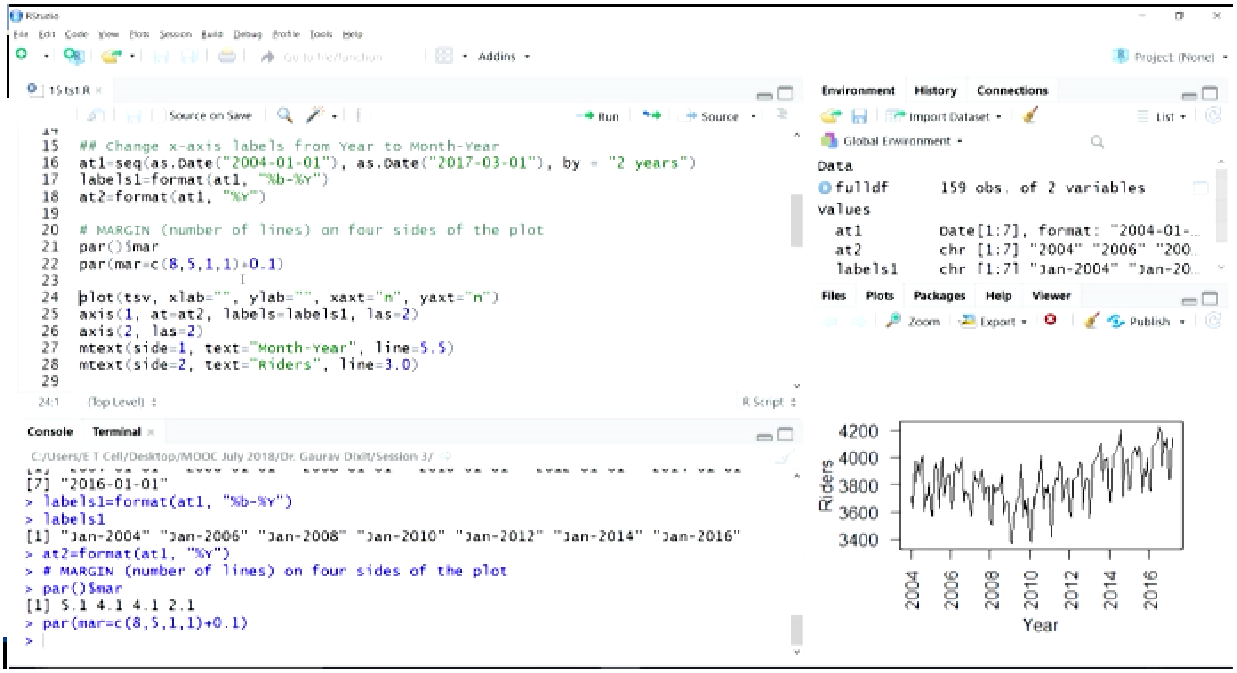
We also specifically talked about how these components can be identified, we talked about different visual techniques that could be used to identify these components, then we started our exercise in R to you know understand these things in a more detail, in a practical manner, so let’s go back to our R code.

So in previous lecture we had created the time plot and we were trying to change the X axis labels to appropriate names, so that it can reflect the kind of time series that we have, the monthly you know ridership, monthly ridership data that we have, so till now you know in the previous lecture we had created this code where we could change the label names and we also computed the tick marks, now what we are going to do is because this labels here are going to be instead of 2004 where we are going to have Jan 2004 instead of 2006 we are going to have something like Jan 2006, so all these years would be change from month year that means Jan 2004, Jan 2006, so these labels would require a bit more space, so therefore we need to change certain parameters, graphical parameters before we can, before we should go ahead and get the plot.



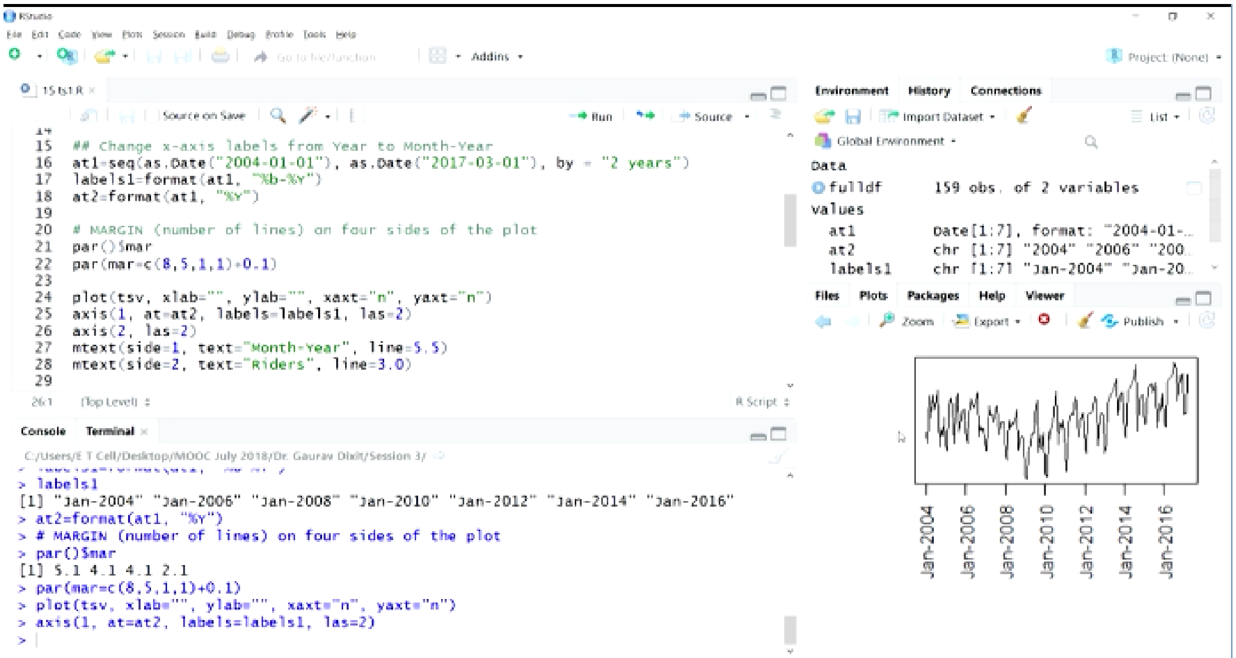
So essentially the margins need to be adjusted for these new changes, changes in terms of labels, so let's look at the current margins, so these are the default margins, so if we use this function par and this particular you know parameter within par is mar that is for margin, so this is 5.1 for the bottom, and then this is, in this clockwise fashion so first bottom then we have the left, then top, and then right, so 5.1 for bottom, then 4.1 for left, then 4.1 for top and then 2.1 for right, so this is the current scheme of margining which is clearly visible in the plot, so we would like to change this because now the labels name would be larger in the bottom section, xl label is going to be larger so which require more margin here.

Similarly we'll also increase margin space on left side also, so bottom and left side we are going to increase the margin, you see the top margin is not being used right now so we'll decrease this so that the width of our plotting box, plotting region you know doesn't you know shrink, so we'll take advantage of this unused margin here on top, and then we'll also like to reduce the margin on the right part of this plotting region, so we need to run this code that you can see here, par and then margin we can specify these number of lines, so essentially this values that we are seeing these are the number of lines so margin has been measured in terms of number of lines on four sides of the plot, so let's run this code and that particular margin would be set for the plotting region, so this is done.



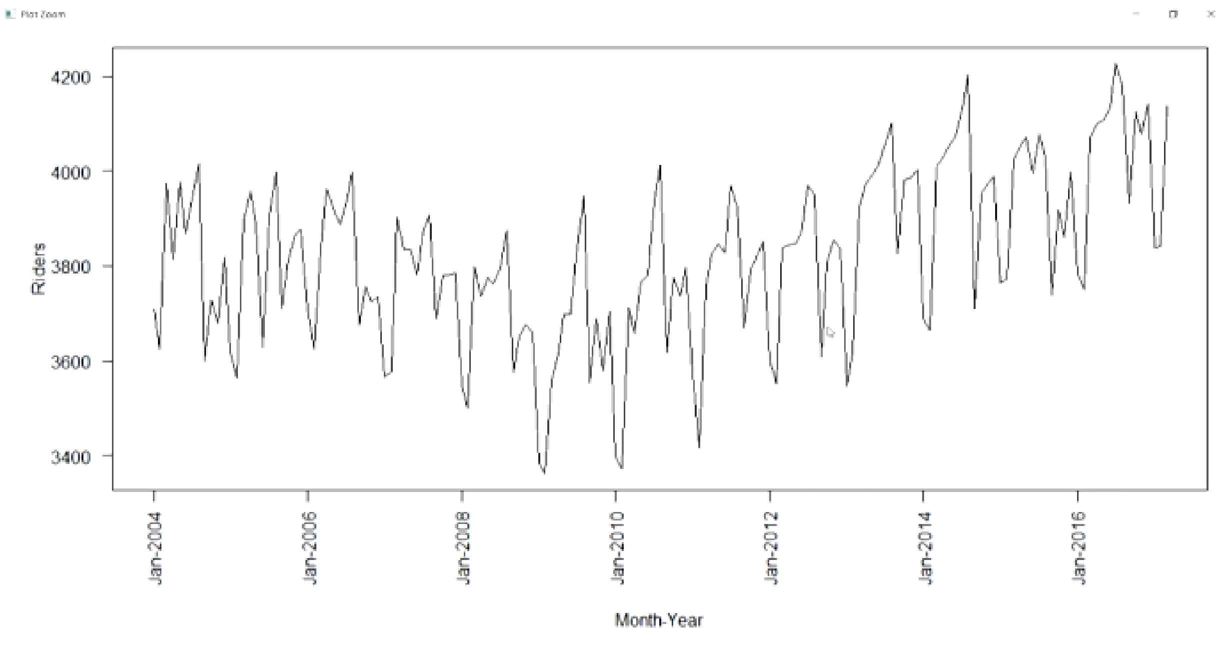
Now we are again going to create a new plot, so let's run this plotting function, first argument as you can see is the time specific vector that we had created, and then we have the labels for Y axis, so which have been kept vacant because we would be using different function as you can see here, M text to label this you know axis and then we have the other arguments XAXT, YAXT which essentially are indicating that the axis labels are not to be plotted because we are going to use these axis function to specifically plot these labels, so different function are going to be used to label these axis, and different functions are going to be used to plot, you know plot these print these axis labels, so let's run this, scaled this plot first, so you can see this plot, now this is more towards the top right section of this plotting region because we need larger labels, so you can see the axis function first argument is one, one is essentially referring to the bottom side of the plot, because we are going to change the axis of the bottom side, so at 2 which we had computed here as you can see, this is essentially capturing the tick marks where the labels are to be plotted.

The next argument has captured the labels which we had created so these labels are going to be printed at those appropriate tick marks specified by this argument, and the labels are specified here, and then the way they are to be printed, so let's run this.



Now you can see, now the labels has changed from just year 2004 to Jan 2004, Jan 2006 and so on, then we have axis 2, 2 essentially means the left side of the plot, so that is to be, so we are not doing much change here, we are just re-plotting that, so you can see here.

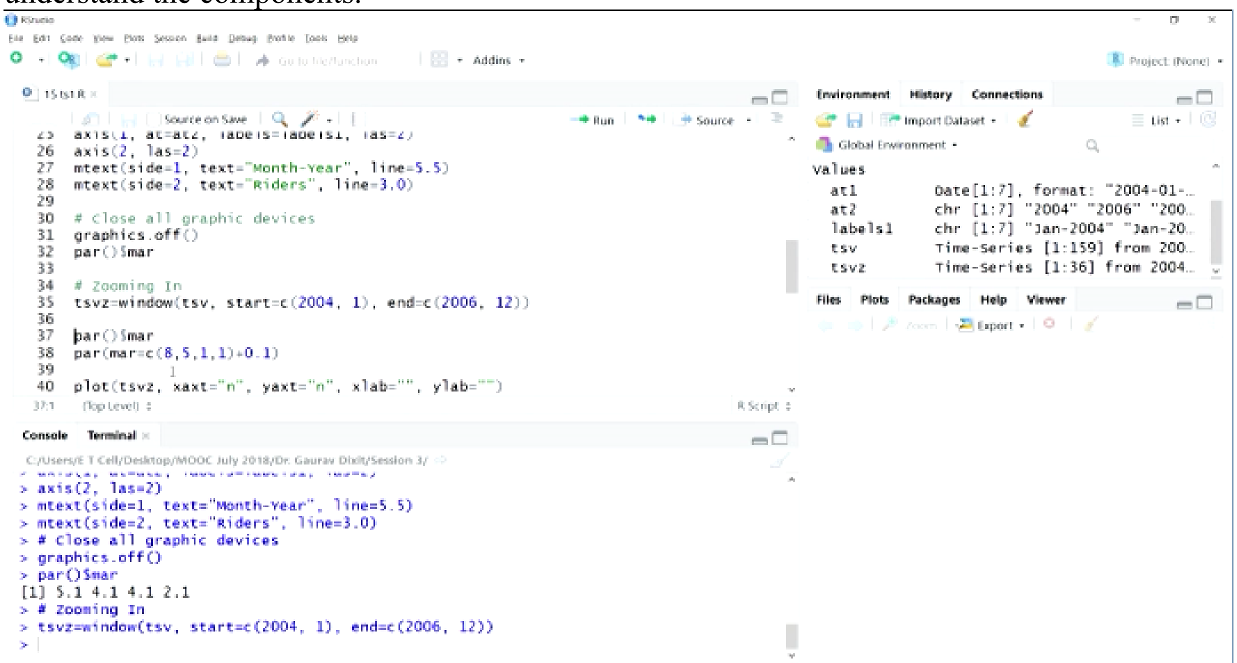
Now we are going to change the you know X axis labels, so now we are going to call it month year, because month information is also offended and in the Y axis side there is not much change, so just riders there, so you can see in this M text function that first argument we are specifying the side which we are manipulating, and then the text that is to be used as the label, and then the third argument is the important one the line, so the margin where this labels are not going, are now going to be printed, so this is important, so we can see 5.5 is so bit, little bit of dry and run, you can also find out the appropriate number of lines to be specified where this labels are to be you know printed, so let's run this, you can see month and year clearly coming there, and then riders.



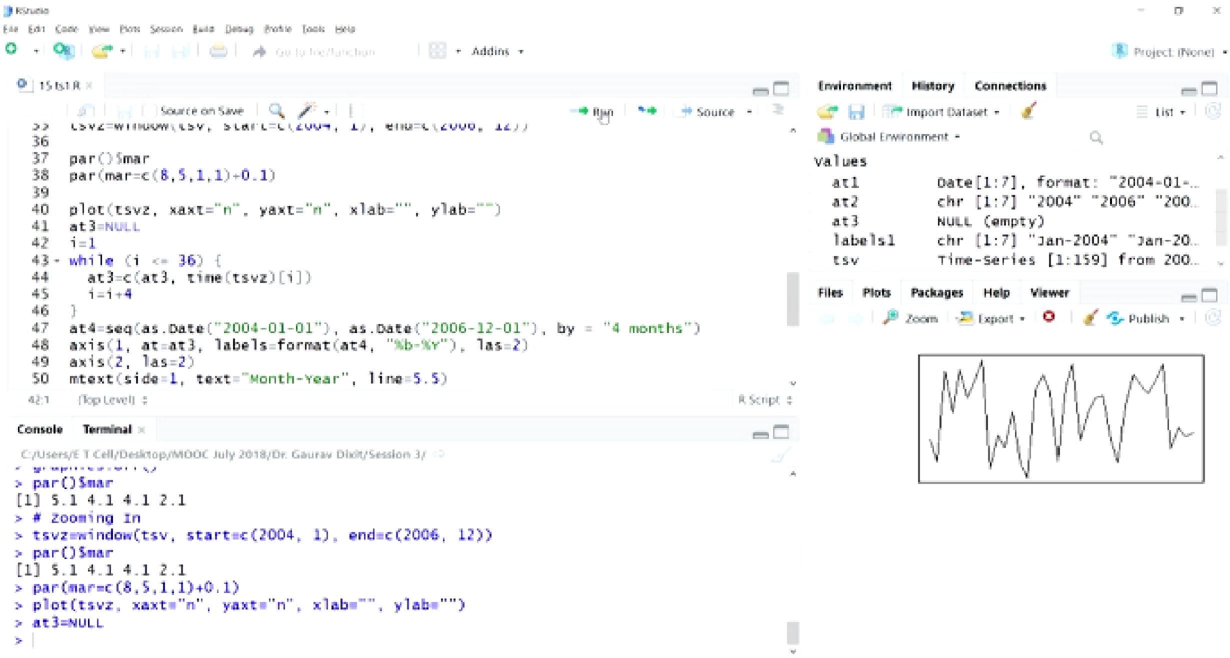
So now this is the plot representing the time series correctly, appropriately representing the time series data that we have, so you can see on the X axis, then time axis we have the you know data riders, data for you know different months and on Y axis we have the number of riders. Now if we look at this particular time plot so from here we can see the trend is something like this, this seems to be you know a polynomial quadratic kind of you know trend here, and if we look at the label, if we look and identify the label of the series you can see most of the values you know this 3800 this seems to be the label of this particular series, because if we look at the overall series here so this particular 3800 seems to be and seem to be capturing the average level of the series for the given period you can see here, 3800 and this line is seems to be capturing the level of this particular series.

Trend we have just talked about, seems to be quadratic term from here they've seems to be slight U shape, small u shape here, so that is the trend, if we look at different short periods of the series, rather quite short period of series, we see you know a particular kind of swing reappearing again and again in the series, right, so for example here we can see this particular swing shape is there and then again it is repeating in many places along the series, so we can see, you know, a peak and non-peak period so clearly we can see peaks and non-peaks period occurring in a cyclical fashion throughout the series, so they are seems to be a seasonal, components seasonality, component as well in this series, so let's close this.

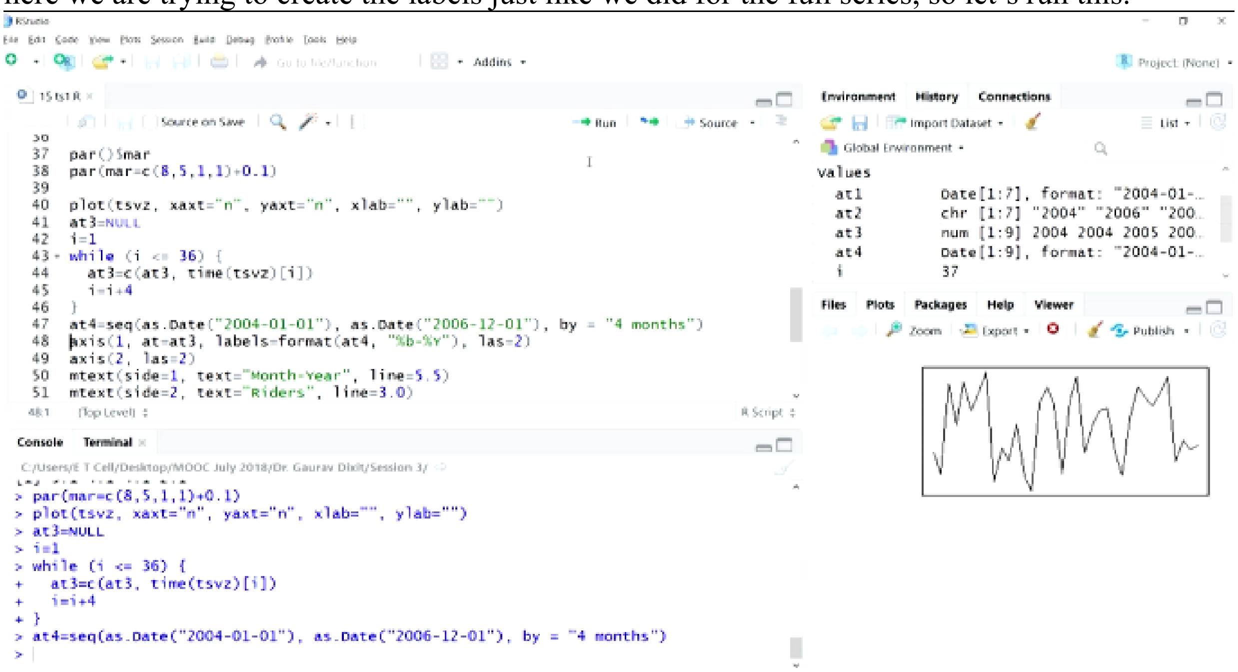
Now there are few more commands that would be sometimes useful, for example graphics.off, so what this particular function would do is we'll close all graphics devices, so essentially all the plots that we have created till now they would be erased, so let's run this. And once we do this let's check the few graphics parameter that we had modified, so you can see the default values are restored after running this command, so let's move to the next point and that we had discussed in previous lecture for identifying the time series component, so this point is about zooming in, so when we, you know we talked about this that when the time series, the specified period is quite long so sometimes some of the you know cyclical pattern might not be clearly visible, some of the patters with respect to seasonality or even trends could be hidden, so what we can do is we can zoom in to a shorter period and then analyze that particular time plot to understand the time series component, so the same process we are going to do in this R studio environment, so this is the function window, so this window function can allow us create a short you know time series using the time series vector TSV that we have, so you can see 2004 the arguments, the first argument is TSV, then the second argument is the start of this window, and the third argument that is end, is the end of this window, so we are starting from 2004 that is start of the series, but after you know 2006 last month that is December 12 we are going to stop, so three years of data is going to be, so we are going to create a window of three years of data from this series, and this window we are going to plot and analyze for different time series components, so let's run this, and you'll see here TSVZ in the environment section a time series object has been created and you can see just 36 values are there because we have taken 3 years data, so every year we'll have 12 data points so in total we have 36 data points for 3 years of data, so this from that overall series first 3 years of data that window we have just created and now we are going to you know look at time plot of this particular part of the series to understand the components.



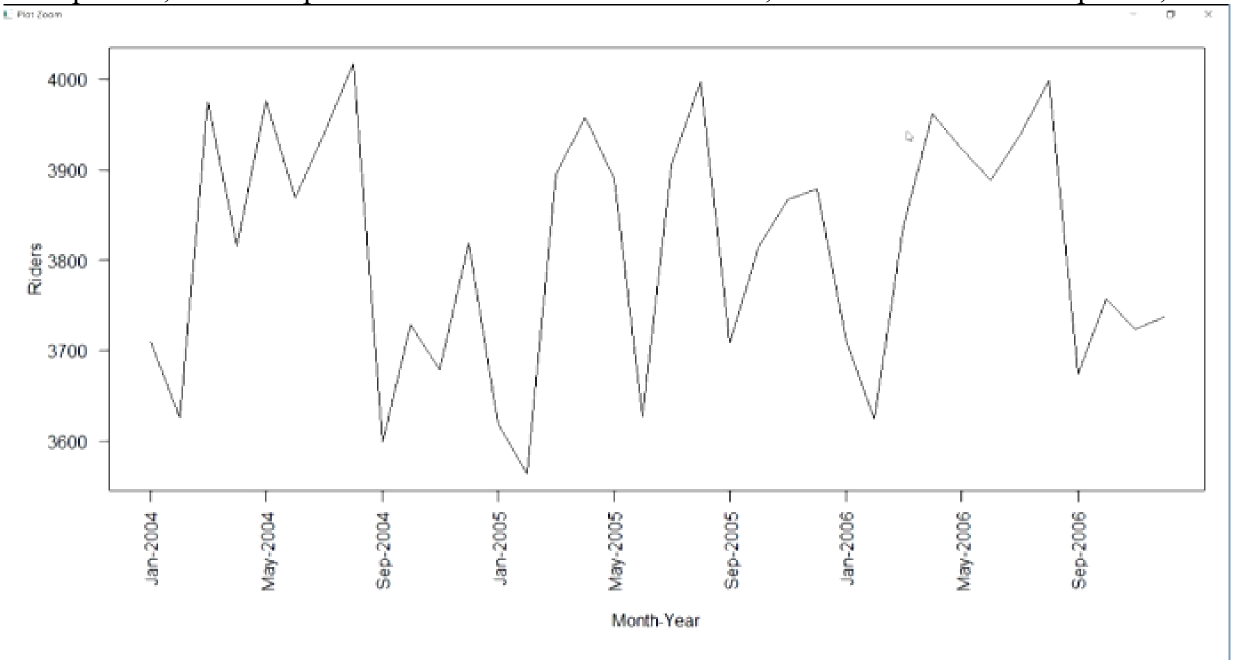
So again we'll do the same process like we did for the full series, we'll change the margins here so that we are able to create the appropriate plot, so default margins are these, then we would like to switch to these margins as we did for the full plot, so now you can see we are using this TSVZ this new window time series vector that we've created, in the same fashion we are going to plot, so the plot has been created as you can see in the top right part of the plotting region,



now we are going to create the labels so for that we require some you know some lines of code here where we are trying to create the labels and tick marks, so you can see at 3 as defined as null, and since we have I is counter here in this while loop where we are running this at 3 to create this vector, so 36 months are there so therefore we are creating that particular vector, so once this is done this we are creating at 4, so at 3 is going to give us the tick marks, so 36 months are there so appropriately we have captured the tick marks and the at 4 we can see here, here we are trying to create the labels just like we did for the full series, so let's run this.

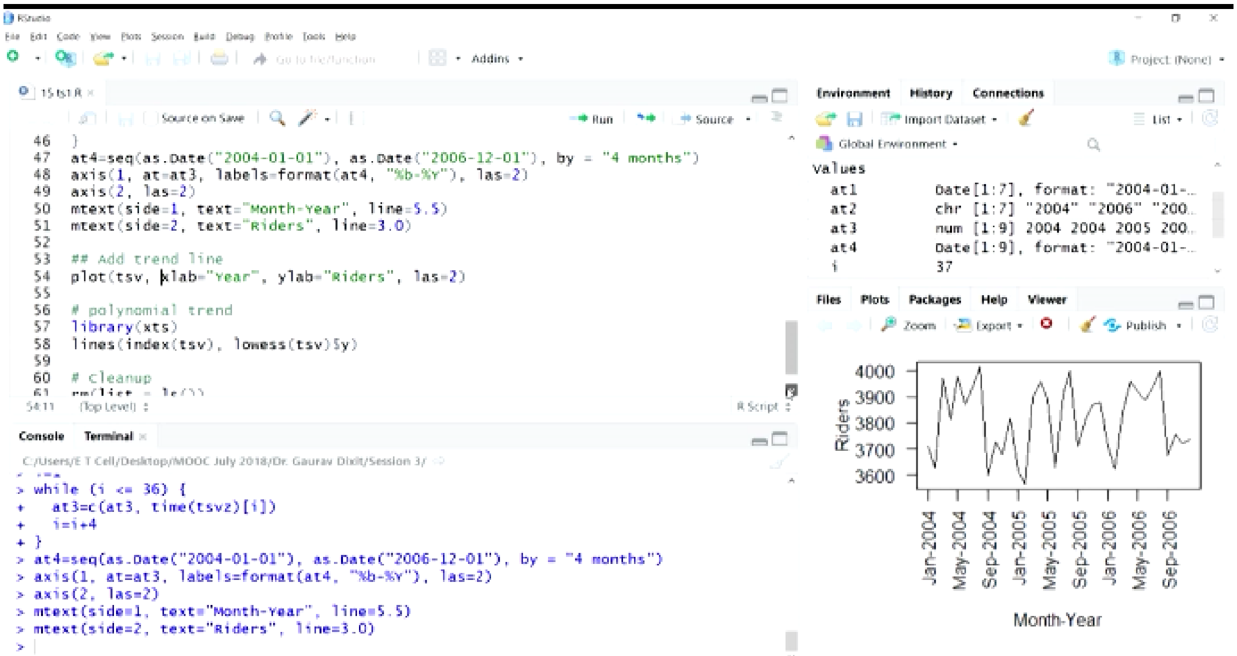


Then again we are calling axis function, so the axis labels are going to be plotted at these tick marks as you can see, you can see at 4 we are using again the format function to make it month year format, so let's run this, you can see now in the plot the month year that information has been printed, now let's plot Y axis and then labels for this 2, so let's zoom in to this period,

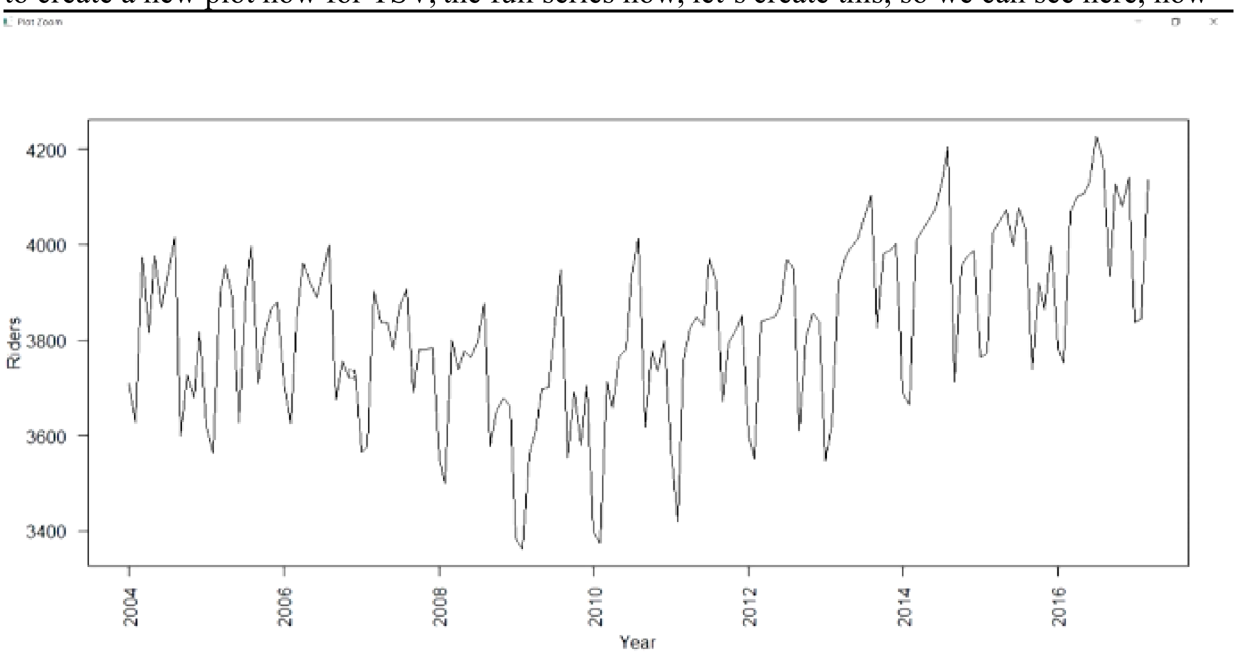


now you can see if we compare this plot, so this plot is just for you know first 3 years and we compare it with the previous plot that we had for the full time series from 2004 to 2017, and we look at this plot, so this plot is more clear you know everything is quite clear in this plot, so this is the effect of zooming in to a shorted period of time, so now we can see the label list quite clearly, specifically in the initial part of the series where we have zoomed in the label is about 3800 number of riders, there about and we can see now in this case if we look at just this part of the series that the trend is seems to be linear, however if we look at the full time series the trend seem to be you know quadratic in nature, polynomial in nature.

If we look at the swings so we can look at for few months the ridership is higher than it drops down then again is goes up and down, so these kind of ups and downs are visible, right, so we can see here, we can visualize some graphs for example in the month of September, we can see the ridership has dropped and again it picks up you can see, around the January period the ridership picks up and again in the month, around the month of May the ridership is on the higher side here also for 2005 also it is about higher you know higher side, so in this fashion we can see that there are clear swings ups and downs, peaks and non-peaks period which can be identified from this time series, so there seems to be some sort of seasonality in this ridership data, we talked about the trend component also and we can also clearly see the level of the series itself, so let's go back.



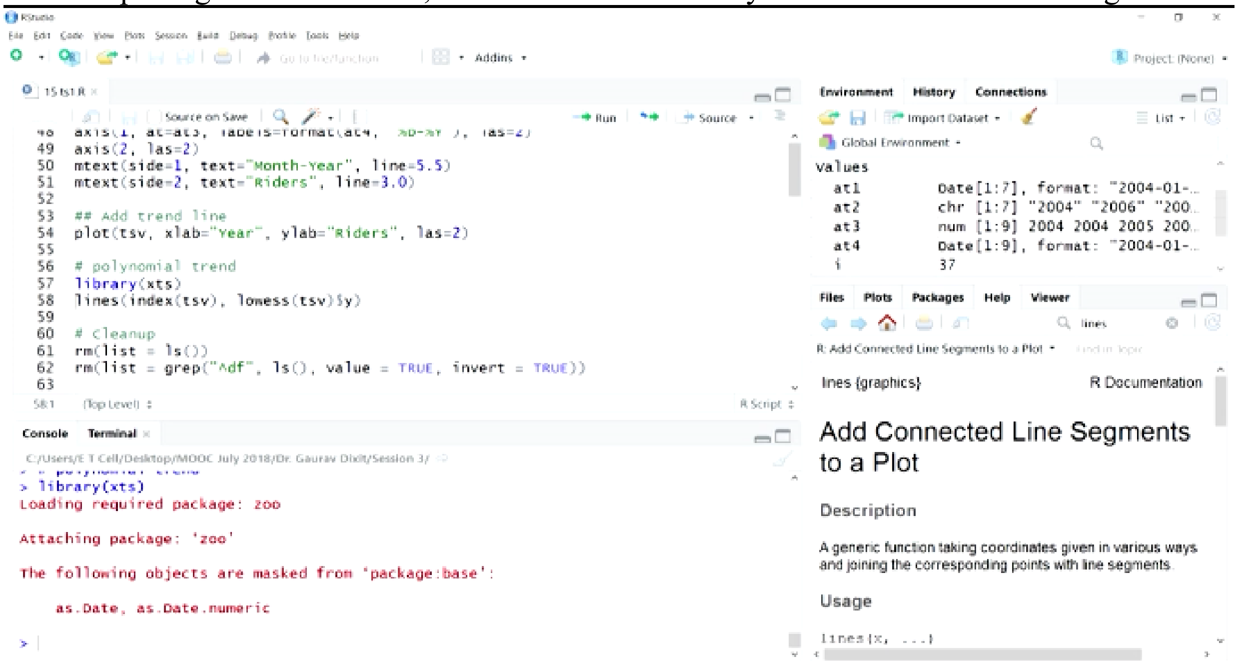
Now let's talk about the adding trend line, so we talked about in terms of identifying the time series components that on top of time plot we can add certain trend line and analyze how they are fitting with the data, so that would also help us in terms of understanding, so we are going to create a new plot now for TSV, the full series now, let's create this, so we can see here, now



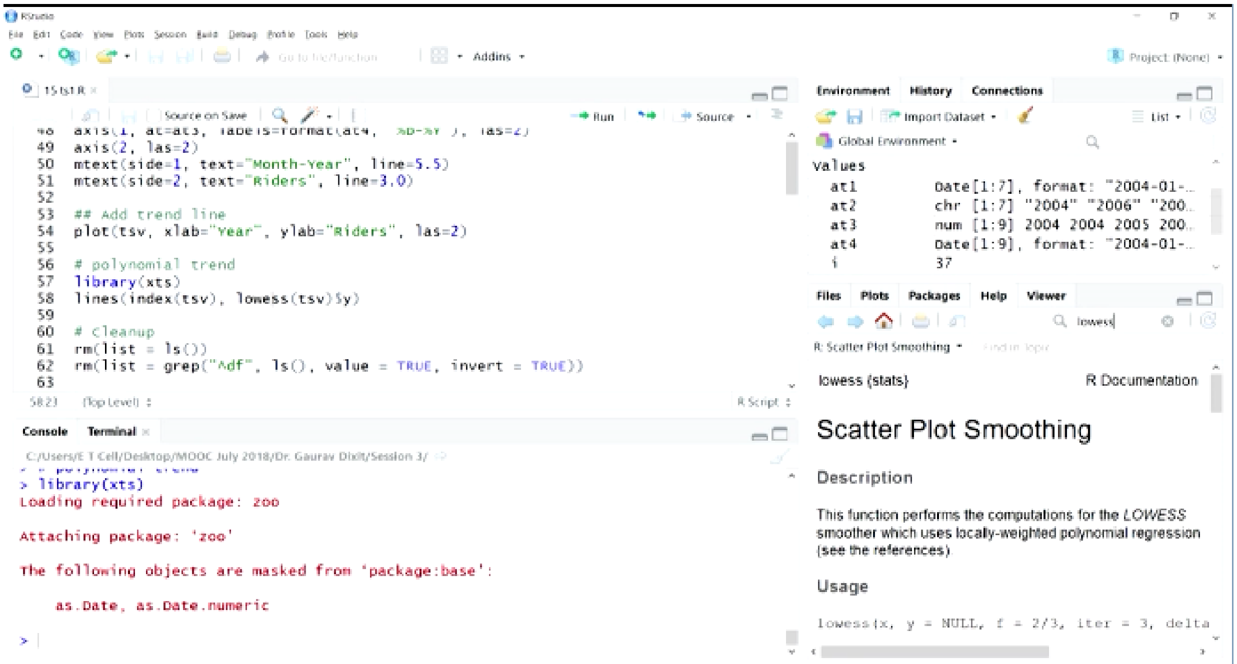
before we do this let's erase all the plots so that we go back to the default margins, let's run it again, now we can see the plot here.

Now as we can see here if we look at this particular plot, we can see here for this given specified period of time series as I talked about, the trend seems to be the polynomial, so what we are going to do is we are going to fit a polynomial curve on top of this plot and see whether it is fitting the data and giving us the trend or not, so for this we are going to load this library

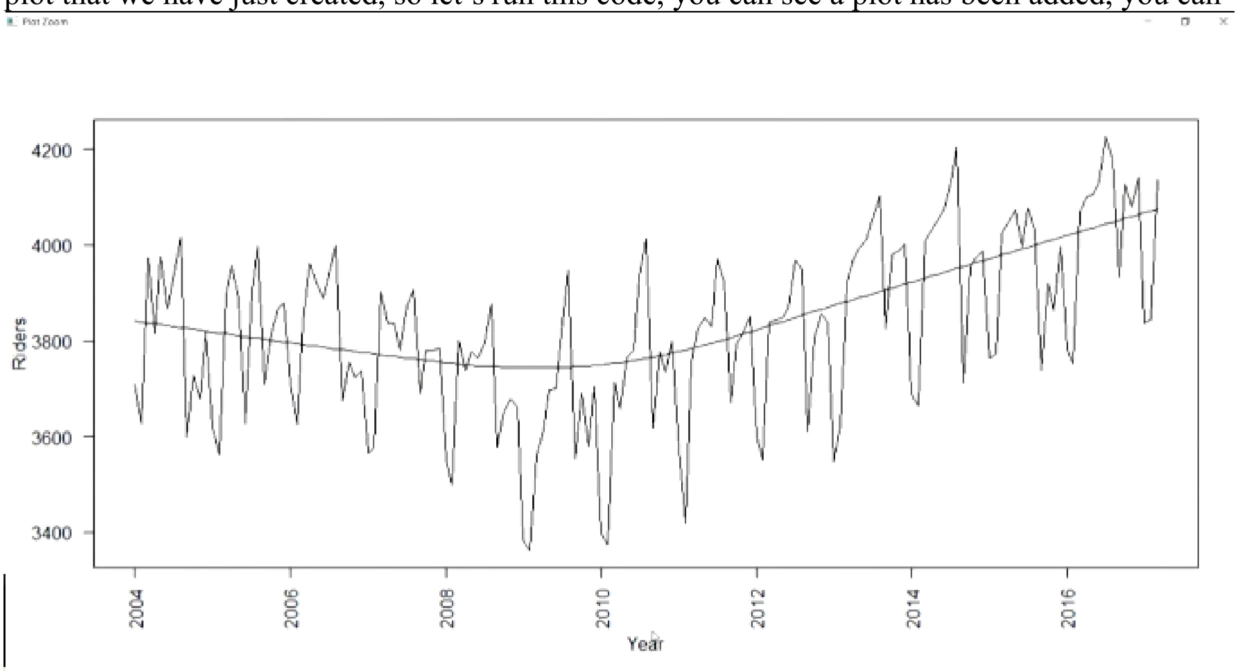
XTS so this is going to be used because we would be requiring this particular function lowess, so let's load this library, so I think we need to first install this one, so let's install this library, installed.packages and then XTS, so let's install this library because we would be using few



function, few functions from this particular package, let's load this library, it's done, now we are going to use this lines function to understand more about lines function, let's have a look at the more detail in the help section, so this is the lines function as you can see, this is part of the default packages which is graphics here, so in the help section also you can immediately identify in the curly basis, you would always see the name of the package, so this function lines is part of the this package, graphics package which is the default package in the R environment, which is graphics, which is one of the defaultly loaded package base packages in R, so if we look at the function this lines is supposed to add connector lines segments to a plot, so if we look at the argument for this function you can see X and Y, so essentially these are the coordinates which this function lines is going to join and create segment, so the same thing we are doing here, so we are again using index function, so this index function is going to extract the, you know the time and X that X axis valuable for us from the TSV, so if you're interested you can find more details about this function index, so we'll get the, so our X axis point are going to be represented by these time index values which we are extracting using this particular function index from this time series vector.

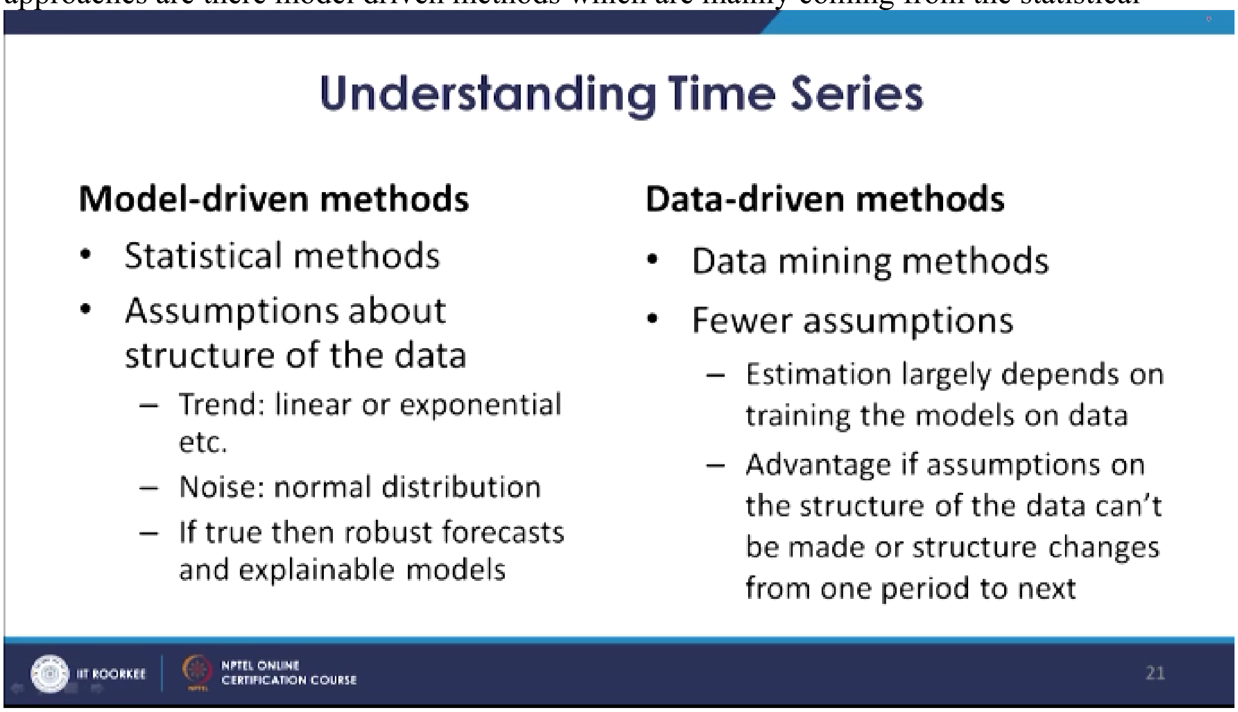


Then we are using the second function lowess this is more important, so this particular function we are using specifically for creating the polynomial curve that we have, so let's look at this function lowess. So this function if we look at this is part of the stats package, so probably we did not require that XTS package as of now, and all this function that we are using they are part of the, already preloaded packages, so lowess if we look at this function this function performs the computation for the lowess smoother, so this is also a kind of smoothing function which uses locally weighted polynomial regression, so essentially what we are going to get is locally weighted linear fit polynomial curve, so this particular curve is going to be plotted on top of the plot that we have just created, so let's run this code, you can see a plot has been added, you can



see a plot polynomial curve has been added, so this particular curve has been computed using the lowess function, but if you look at the curve so this seems to be polynomial in nature, so this is what we were talking about, so the trend of this particular series seem to be polynomial in nature, and when we zoomed into a, zoomed into first 3 years it looked like you know linear but if we look at the entire time series this seems to be polynomial where this curve that we have just created added on top of this time plot, it seems to be the fitting, it seems to be fitting to the data.

So some of the points that we talked about in terms of identifying different time series components that we have been able to cover, now let's go back to our discussion, further discussion on time series, so now let's talk about the different approaches that could be used to model time series, so as we have done in previous course also, typically two types of approaches are there model driven methods which are mainly coming from the statistical

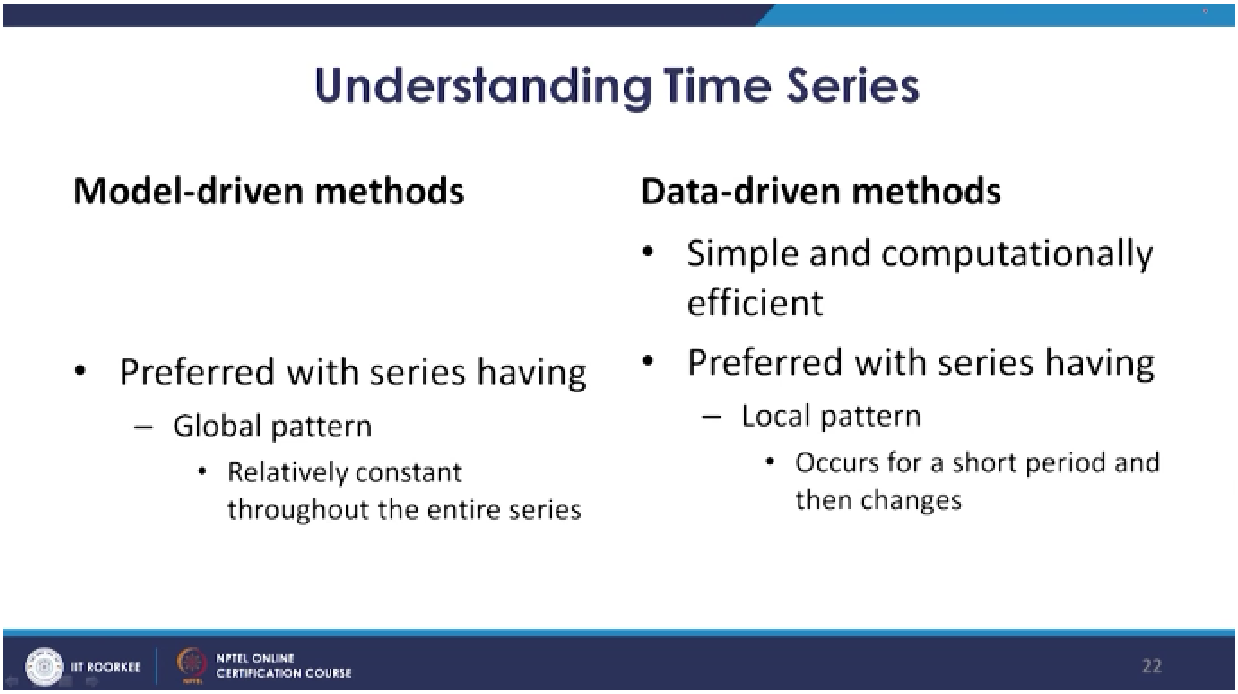


domain, and then we have data driven method which are mainly coming from the computer science data mining domain, so the similar kind of you know just like we you know the discussion that we had for the you know cross sectional analysis in the previous course which is mainly about applying supervise learning techniques or unsupervised learning techniques.

Similarly for time series also we have some approaches coming from model driven methods, some techniques coming from data driven methods, so let's understand the difference between these two approaches, so model driven methods, first point as I have talked about, statistical methods mainly data driven methods typically data mining methods are used, if we talk about the model driven methods so just like any other statistical method they are going to be assumptions, so assumptions about structure of the data, just like in multiple linear regression, we assume the relationships to be linear and therefore that is model in the multiple linear regression equation in the same fashion, similarly here also we are going to make certain assumption about structure of the data and those assumption are going to be used in the model equation, so for example trend so typically we are going to make this assumption whether the trend is looking like linear or exponential, so that is why the identification of time series component that we just did in R studio it becomes more important, so if we are able to identify what is the kind of trend that is looking there from our visual inspection, so that can be used in our modeling process, specifically if we are using model driven methods.

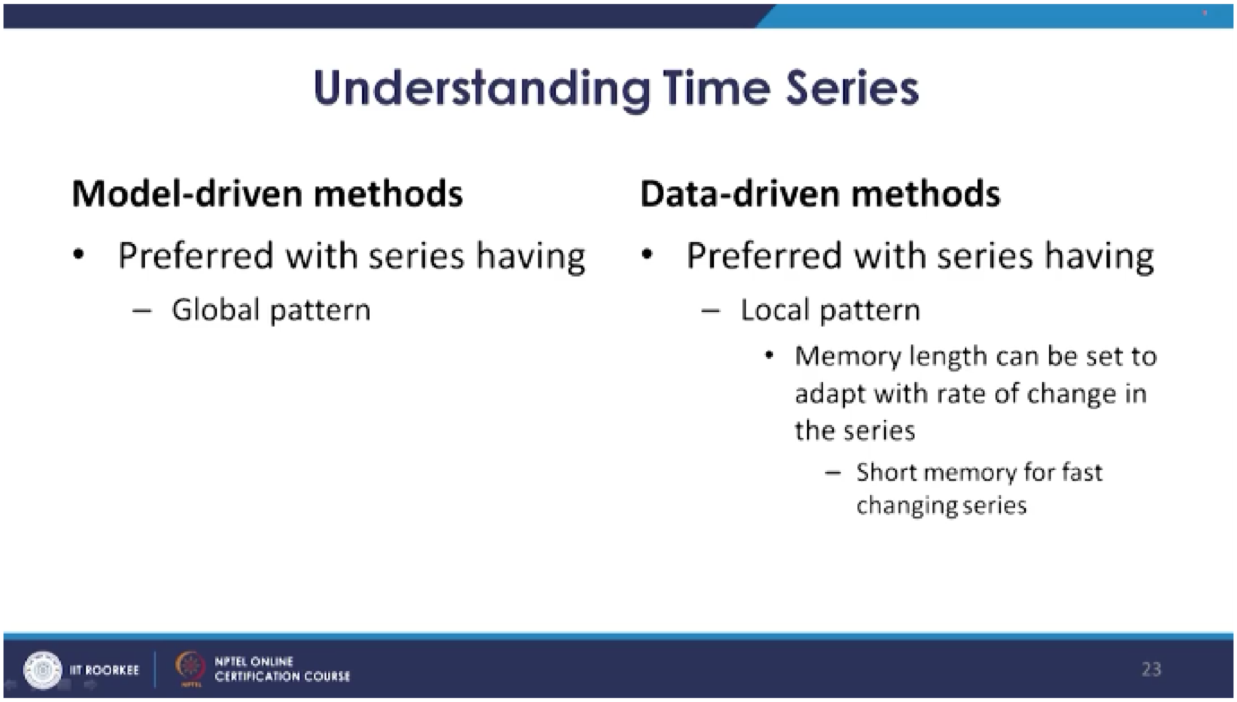
So trend, linear or exponential if we assume that is the structure in the data, time series data, then that is to be used, so whenever we use this kind of structure then of course certain restrictions are also going to be there. If we talk about noise then also the statistical method assume this you know this particular structure that noise is going to follow normal distribution, so this is also about an assumption about the structure of noise, right, so it is assumed that the trend the form that we are, we have used to model the data that is going to capture the most of the points and the noise part is going to be left with random variation, so that is going to be again distributed following normal distribution, so these are certain assumptions that could be involved in statistical method, so one advantage of some of the assumption are if these assumption are you know met in the data then the kind of forecast that we are going to get, or going to be more robust,so certain changes in the series, certain changes in the period of the series are the observations or values taken by the series variable, the forecast is still going to be robust to those changes, explainable model so because of the you know we are assuming about the structure of the data, so the relationship can be quite clearly specified, can be clearly explained, so these are some of the advantages, so we make certain assumptions, but if the assumptions are met then we have robust forecast and the modest can also be explained.

If we talk about the data driven methods, so there fewer assumptions are involved because the whole modeling process that largely depends on training the models on data, so as you can see here, in the first point within fewer assumptions, the estimation largely depends on training the models on data, so the learning is typically based on data, so therefore fewer assumptions are involved, we don’t make any you know specific assumption about the structure of the data and incorporate that in the modeling process, rather the raining process itself the modeling process itself depends on learning from the data, so these particular methods they are advantageous if assumptions on the structure of the data which we typically use for model driven methods, statistical method if they are, if they can’t be you know, they can’t be made or you know if there are any such, if there is any such structure present it keeps on changing from one period to next, so therefore if the assumptions that are involved, if they’re violated, if they can’t be made then probably data driven methods are more suitable to model time series.

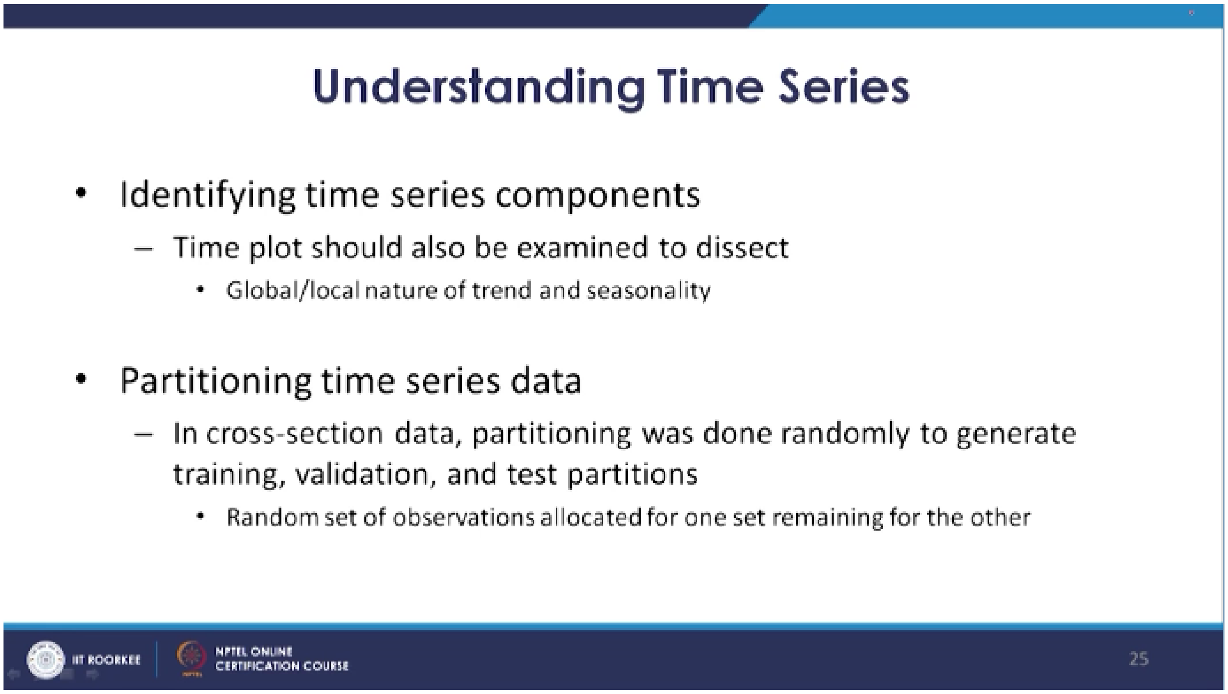


Few more points about comparing these types of methods, so if we look at model driven methods and data driven methods, data driven methods are more simple and computationally efficient, because they are going to be based on you know, computationally efficient algorithms and that makes them quite simple and efficient, so that is another advantage that is there with data driven methods.

Let's move forward, so while comparing these two types of method there is another important aspect which is you know when the model driven methods should be preferred, and when data driven methods should be preferred, so if model driven methods are preferred when we have a series having global pattern, what we mean by global pattern, so this is when we talk about pattern essentially we are talking about the you know more often they are not talking about the trend component of the time series and also sometimes we might be including the seasonality component, so when we talk about the global pattern if you know the series is having global pattern and global pattern essentially means is the pattern is relatively constant throughout the entire series, so that pattern is clearly present in the whole series, so we refer that as a global pattern. If we compare this with the data driven methods, so these methods are preferred when the series is having local patterns, so that means these patterns they occur for a short period of time and then changes, so as I talked about that you know just like the statistical method, the model driven method they are preferred when there assumptions about the structure of the data are met, and if they are not met then data driven methods are preferred, similarly if the series is having you know from the same we can reduce that, you know pattern if it is globally applicable then again the model driven methods are going to be you know preferred, these patterns keep on changing, then of course data driven methods are going to be preferred.

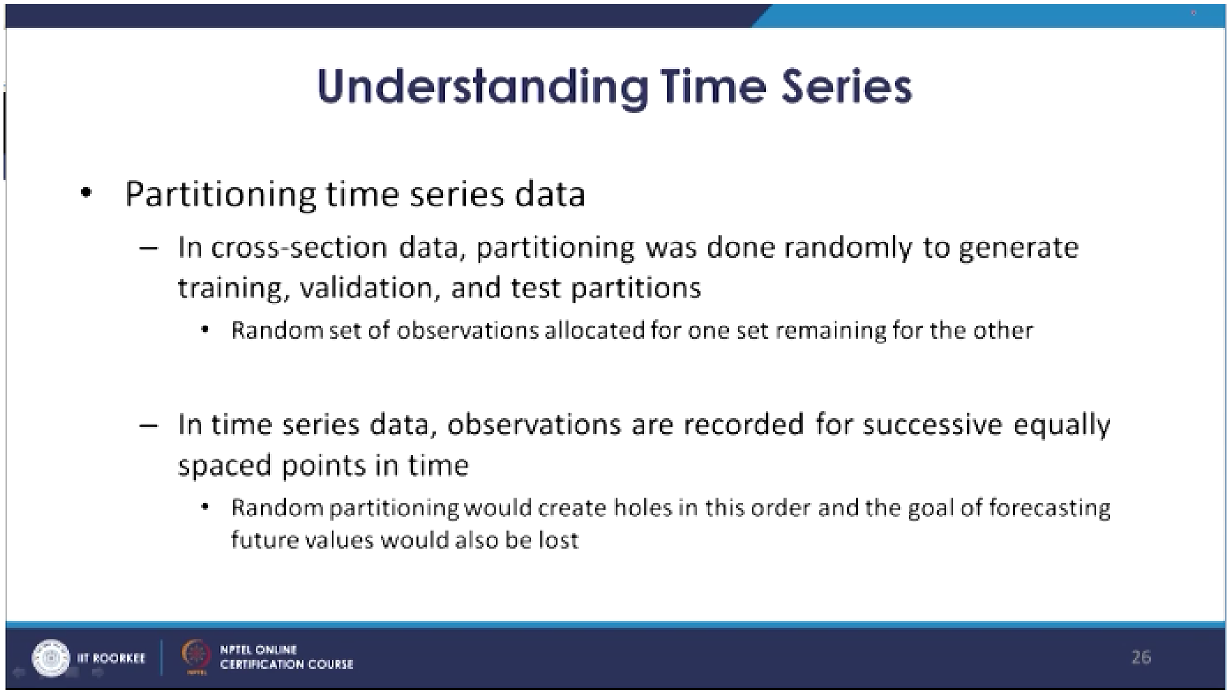


So if we talk about the local pattern, so there also, there can be certain scenarios, so to capture these you know changes in pattern we typically use certain memory for the model, so this is called memory length which can capture you know how often these trends are going to change, so this memory length also sometimes have to be adjusted to adopt the rate of change in the series, so we typically requires short memory length for fast changing series, because of the trend is changing quite often then the model has to forget the previous learnings you know from the nearby data points and adopt two learnings from the newer data points, and this rate of change if the, it's first then the memory length should be smaller, so this is one, however if the series, even though the these local patterns are changing but this rate of change is slow then probably we can have long memory to have you know influence of a bit large number of points on the modeling and forecasting process.

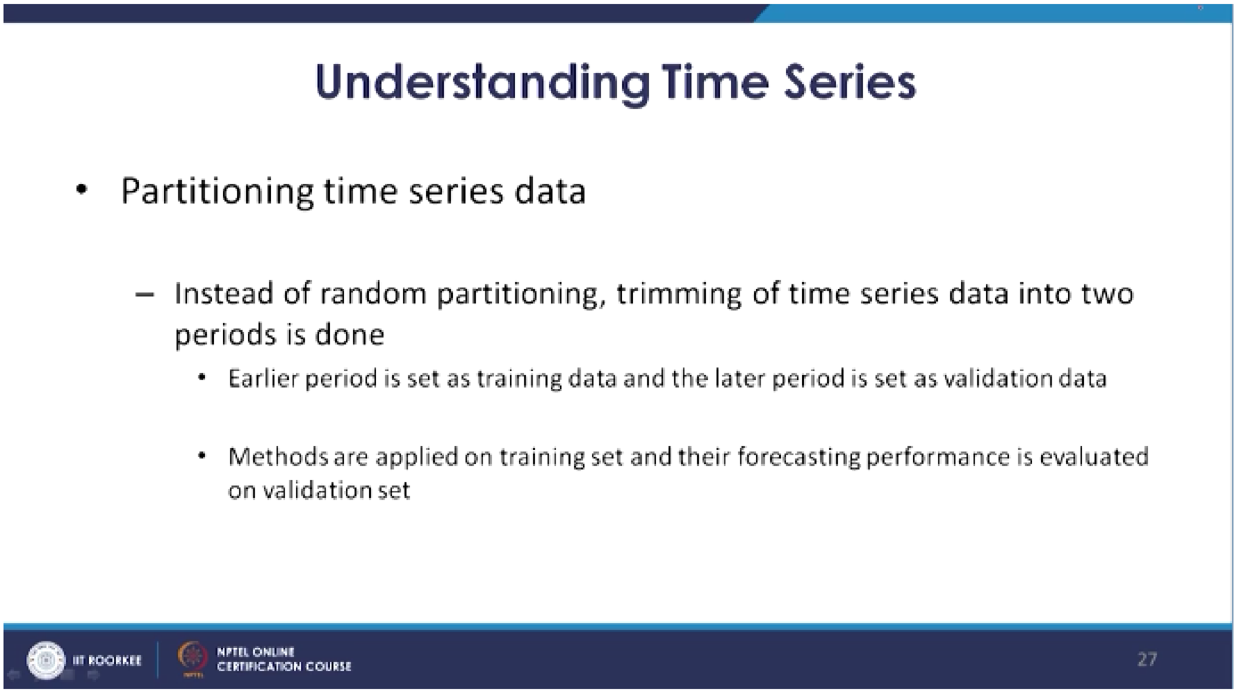


So let's go back to what we discussed in previous lecture also identifying time series components, so in the context of the discussion that we just had, the comparing model based method and data driven methods, and we specifically in the last point we talked about global and local you know patterns, so time plots that we talked about in terms of identifying time series components they can also be examined for the presence of global or local nature of trend and seasonality components, because once we are able to understand whether if the global you know pattern is present, then probably we should use statistical method, model driven methods if local you know trends, local patterns are you know there, then probably we should go with data driven method, so it would be easy for us to select which kind of method we should be using for our modeling and forecasting

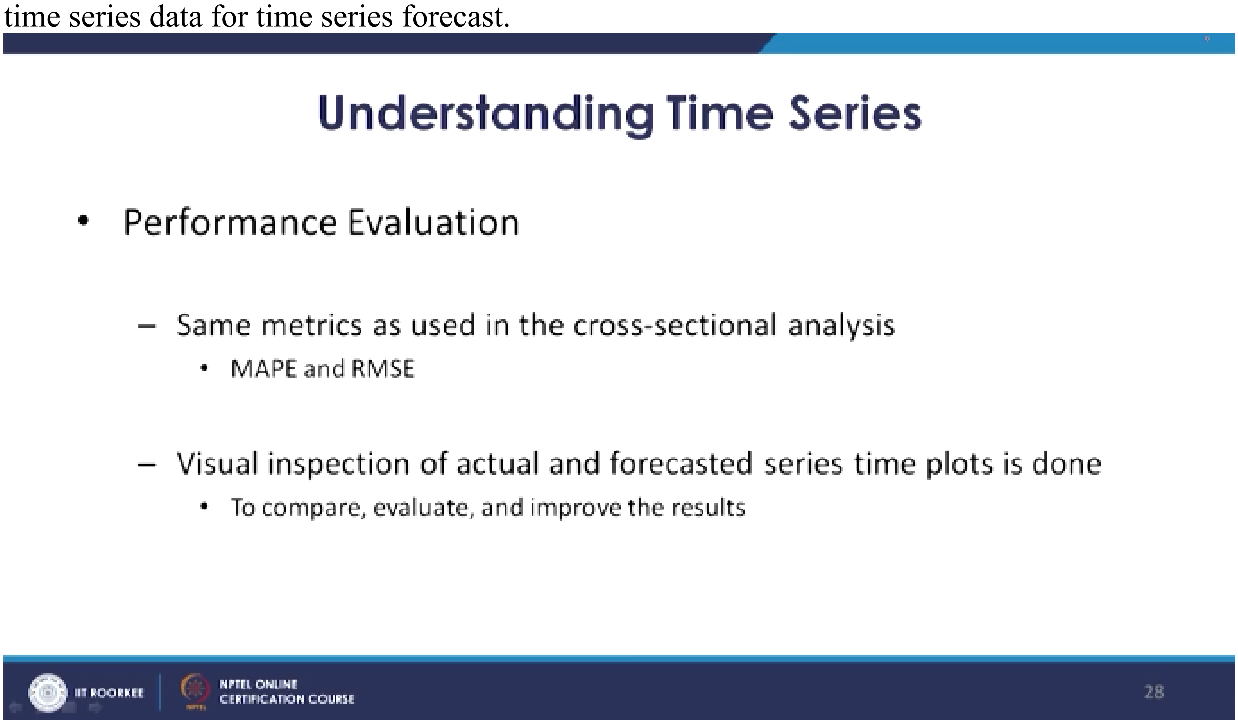
Let's move forward, next point about time series is partitioning, so partitioning time series data this is quite different, quite different to what we have discussed in previous course and also in this course, in the other module, so in cross section data partitioning was done randomly to generate training, validation, and test partition, so these random set of observation were allocated for one set and remaining observation were left over for the other set, so this was the typical routine, because as we understand the now that cross section data typically we collect observations on R variables in one you know at the same point of time, it is one snap shot so because of that you know random set of observations makes sense, can be taken and it will also give us that you know remove the bias that could be there in the convenience and other types of sampling, so this is what we do in cross sectional data, however if we talk about time series data, the observations are recorded for successive, equally space points in time, this is



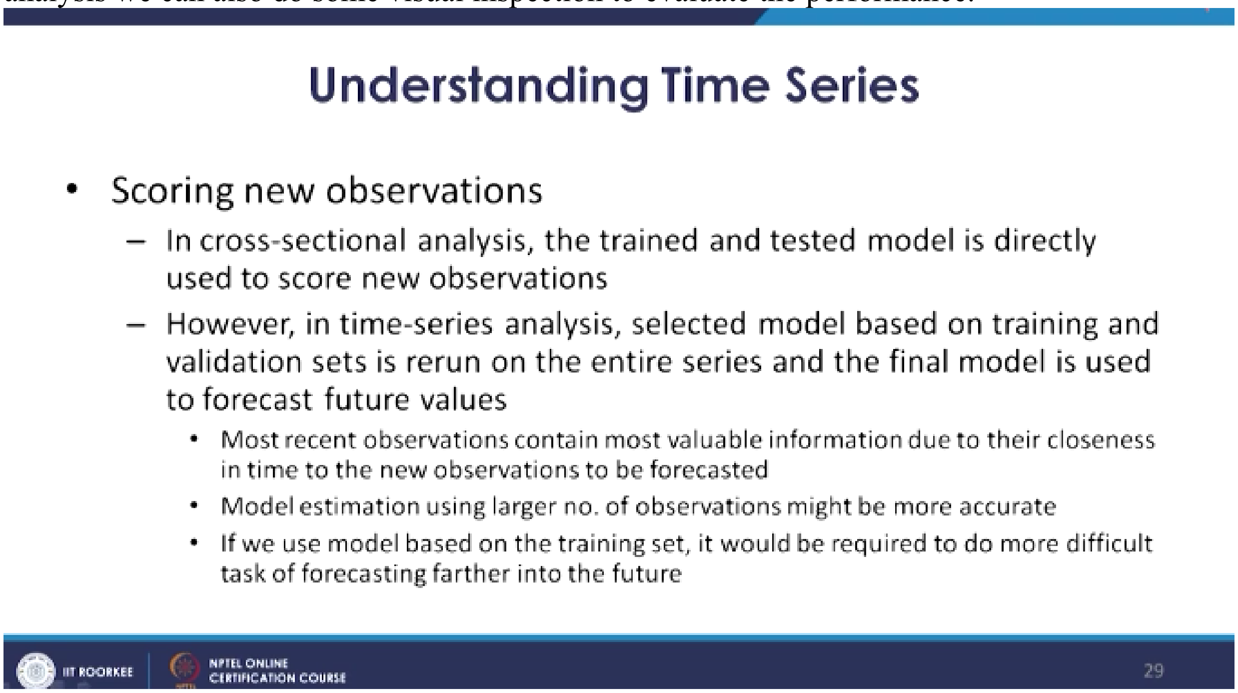
something that we have discussed, so if we do, if we do random partitioning so what it is going to do it will create holes in this order because all the observations they are you know successively equally space points in time, so you record you know for example you record GDP value for a particular quarter, then second quarter, and you know so on, so if we do random partitioning, if some of the you know points have you know being randomly allocated you know from one period and some points might be randomly allocated from other period, so in between they are going to be holes while the time series, the order is quite important, and the purpose of the goal of forecasting future values would also be lost, because essentially in time series forecasting we are looking to forecast future values, so if we do random partitioning these holes and the model that is trained using randomly partition data would actually defeat the main purpose.



So in time series instead of random partitioning we actually trim the time series data into 2 periods, now earlier period is set as training data, and the later period is set as validation data, so this training data period, the earlier period which is used as train data is used to train the model, to build the model, and the later period is actually used to evaluate the performance of the model, so as you can see the next points, so methods are applied on training set and therefore casting performance is evaluated on validation set, so instead of random partitioning which we use in cross sectional data, cross sectional analysis we use you know trimming of time series data for time series forecast.

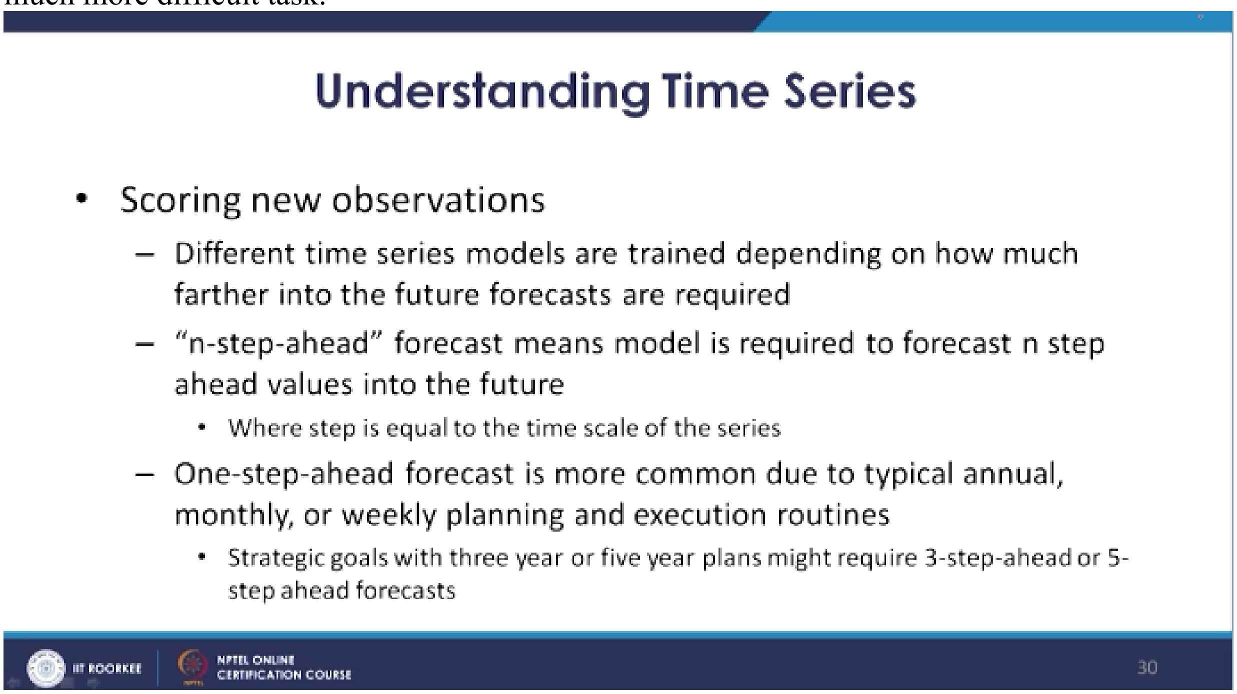


Now if we talk about the performance evaluation, performance evaluation is quite similar to what we do in cross sectional analysis, so the same kind of matrix RMSE MAPE can be used here also, however as we have talked about visual techniques they are quite important part of the overall time series forecasting and analysis as well, so here also in terms of performance evaluation also visual inspection can be helpful, so actual and forecasted you know series time plots we can create and we can compare them how the forecasted time series values are behaving and how they are changing, so that kind of comparison evaluation and based on you know the gap that is there between the actual series and forecasted time series we can always improve the results also, so this kind of apart from the matrix that we are used in cross sectional analysis we can also do some visual inspection to evaluate the performance.



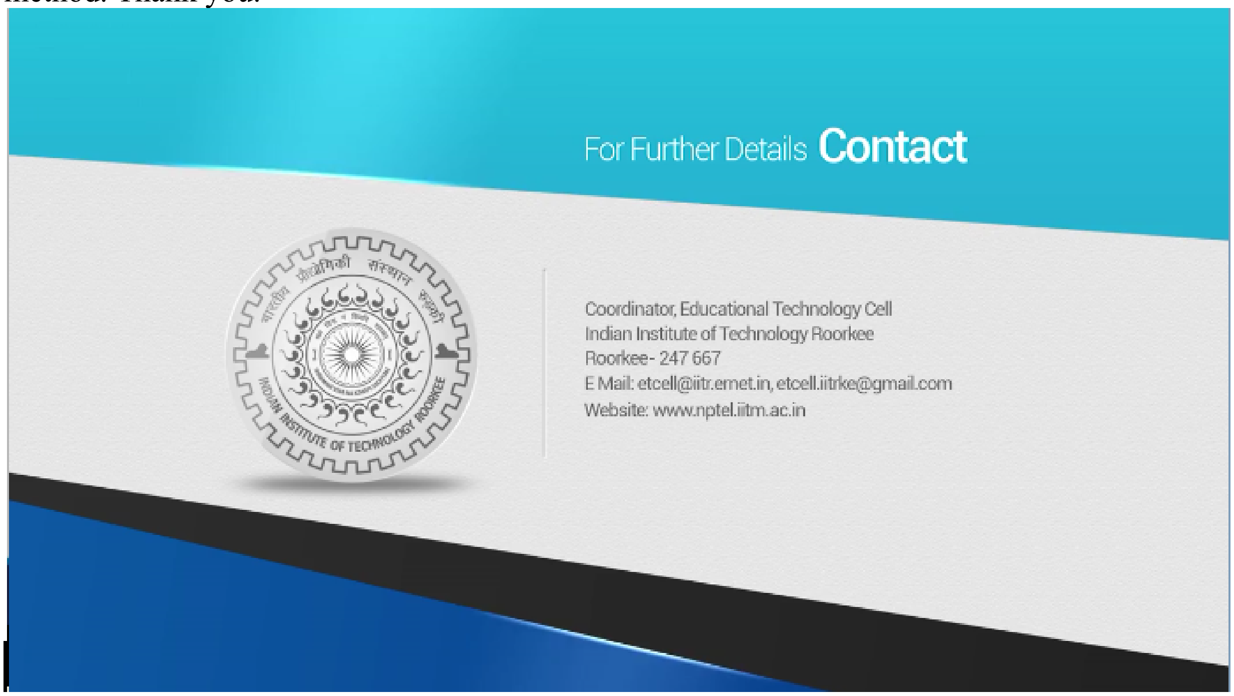
Now let's talk about a scoring new observations, so in cross sectional analysis once the model has been trained and tested it can be directly deploy, directly used to score new observations, however the same thing is not really recommended in time series as you can see in the second point, in time series analysis selected model based on training and validation sets is rerun on the entire series and the final model is used to forecast futures, future values, why this is done? Let's understand so some of the points are discussed here, so instead of using the series which is there, which have been build using trend data we again you know that is typically used to select a model, select a particular technique which is performing well, and then that particular you know model is rerun on the entire series, so and this final model is typically used to forecast future value, future values, so let's look at few points why this is done, so most recent observation contain most valuable information due to their closeness in time to the new observations to be forecasted, and therefore when we are looking to forecast future values, if the entire series data is being used to build the model and then you know forecast future values, the most recent observation would also be the part of the modeling exercise and therefore contain, will contain will have the leverage of this most recent values and therefore most valuable information, and might provide better forecast, so this is one reason, why this kind of thing is done.

The second is quite obvious that model estimation using larger number of observations might be more accurate, because if we restrict ourselves to just trained data observations the number of observation could be fewer, but if we use the entire you know series then the number of observation would increase and that would give us you know chance to get a more better model, to build a better model.
Third one is if you use model based on training set it would be required to do more difficult task of forecasting further into the future, because you know the training partition if the model is build on using just those points then you know and we use it to forecast future values then the observation which are part of the validation data, so those points are there and then we are going to forecast points which are coming after that, so therefore the model which is build on training data will have to forecast beyond the points which are there in the validation dataset, and the model is going to be required to forecast much further into the future, which would be much more difficult task.



Few more important points about you know scoring new observations, so as you can see different time series models are trained depending on how much further into the future forecast are required, so if we are required to forecast just the next year's GDP value so we would have to build one model, if we are required to forecast you know next to next year's GDP value then we will have to train a, build a separate model, so this kind of thing is typically done, so the same thing is reflected here also, and step ahead forecast means model is required to forecast n step ahead values into the future, so if we have used observation till today in building our model and we are required to forecast you know two step ahead, let's say two step ahead that means not tomorrow, day after tomorrow's values then the model has to be trained for that, so and that is to be used for forecasting. For example you can see the third point also, one step ahead forecast is more common due to typical annual, monthly or weekly planning and execution routines, so typically the way planning and execution happens in firms belonging to different industries either they would prompt for the next year or next month or next week and therefore you know next one step ahead that means next week forecast or next month's forecast or next day's forecast, or next year forecast would be more practical for them and that makes one step ahead forecast on a more common, however for strategic goals and strategic planning 3 year or 5 year you know planning that could actually require 3 step ahead or 5 step ahead forecast as well.

So this was all about understanding time series, time series component, different methods, approaches, and how partitioning and scoring new observations, performance, valuation, all those things are different from what we have done in other modules in cross sectional analysis, so with this we'll stop here, and in the next lecture we'll start with regression based forecasting method. Thank you.



For Further Details Contact
Coordinator Educational Technology Cell
Indian Institute of Technology Roorkee
Roorkee – 247 667
E Mail:-etcell@iitr.ernet.in, iitrke@gmail.com
Website: www.nptel.iitm.ac.in

Acknowledgement
Prof. Ajit Kumar Chaturvedi
Director, IIT Roorkee
NPTEL Coordinator
IIT Roorkee
Prof. B. K Gandhi
Subject Expert
Dr. Gaurav Dixit
Department of Management Studies
IIT Roorkee
Produced by
Mohan Raj.S
Graphics
Binoy V.P

Web Team

Dr. NibeditaBisoyi
Neetesh Kumar
Jitender Kumar
Vivek Kumar
Dharamveer Singh
Gaurav Kumar
An educational Technology cell
IIT Roorkee Production