

INDIAN INSTITUTE OF TECHNOLOGY ROORKEE  
NPTEL  
NPTEL ONLINE CERTIFICATION COURSE  
Business Analytics & Data Mining Modeling  
Using R – Part II  
Lecture-18  
Regression Based Forecasting Methods– Part II  
With  
Dr. Gaurav Dixit  
Department of Management Studies  
Indian Institute of Technology Roorkee

# Business Analytics & Data Mining Modeling Using R - Part II

## Lecture-18 Regression Based Forecasting Methods-Part III



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 few lectures we have been discussion Regression Based Forecasting Methods, in particular in the last part, last you know part of previous lecture we were able to model trend and seasonality both the components, we also looked at the residual plots and so that the final model that we had, that we build was able to adequately capture most of these components.

## Regression-Based Forecasting Methods

- Modeling additive seasonality
  - If this categorical variable has  $m$  seasons
    - $m-1$  dummy variables are created to be included as predictors in the regression equation
- Modeling trend and seasonality
- Open RStudio

So let's move to our next important aspect of regression based forecasting, so there are certain issues when we use ordinary regression models in time series context, so let's discuss some of these issues and how we can overcome them.

## Regression-Based Forecasting Methods

- Issues with ordinary regression models in time series context
  - Observations are assumed to be independent
  - This assumption is typically valid for cross-sectional data
  - However, for time series data
    - Neighboring observations tend to be correlated which is also referred as autocorrelation
  - This information can be used to improve the forecasting models

So one important thing is that observations are assumed to be independent, so whenever if we are applying a you know ordinary regression model this is one of the underlying assumption in all the you know multiple linear regression models that observation the cases are assumed to be independent, however if really look at the time series context, this assumption is typically not valid, so this assumption is typically valid for the cross sectional data where we are gathering information on you know different you know different rows along different observation rows so

which is on a each unique you know subject, it could be individual form or you know anything else, so those could be different so all those instances of that subject, those individuals are going to be different, so this particular assumption is valid there, however when we talk about in the time series context, since we are measuring the same instance of the subject, it could be a particular for example in this, the examples that we have been following here, the ridership so the you know that riders, number of riders in a month that is something that we are measuring and at different you know sequentially you know equally space points in time, so we can see you know these observation, one observation at time  $T$  and the other observation at  $T-1$  or  $T+1$  cannot be clearly called as you know independent, so they depend on you know previous observation in the sense that either the you know for example we talked about the ridership, the number of riders will go down or up in some sense so the historical values, previous values and you know the future values so they are dependent on each other, so this is the point precisely the point that has been mentioned in the slide as well, so you can see here third point, however for time series data neighboring observations tend to be correlated which is also referred as auto correlation.

So this is the particular aspect of time series that we are going to discuss now in this particular lecture, so neighboring observations they tend to be correlated, so ridership in the coming month number of riders that are going to be there in the campus in the coming month might also be in a way determine by the number of riders which are right now in the current month, so these dependency is very clear and a straight forward, so this has to be accounted in our time series forecasting, so as we talked about issues with ordinary regression models so this particular aspect is not accounted for.

## Regression-Based Forecasting Methods

- Autocorrelation
  - Correlation between values of a time series in neighboring periods
  - Describes the relationship of series with itself
- Computing autocorrelation
  - Correlation between the series and a lagged version of it
  - A lagged series with lag 1
    - Values of original series moved forward by one time period to occupy one-step-ahead time point in future
      - In effect, time index of original series moved back by one time period

So as you can see in the last point in the slide this information can be used to improve the forecasting models, now we came across the stem autocorrelations which is about neighboring dependence of neighboring values in a time series context specifically, so let's understand what is autocorrelation in a bit more detail, so correlation between values of a time series in neighboring periods, so in a time series data the neighboring values tend to be correlated and this correlation is actually referred as autocorrelations, so this in a sense autocorrelation the way

the name is being used, this equation describes the relationship of series with itself, how the series, how the values when we say the values of a time series you know in the neighboring periods they are correlated, so in a sense series is, a particular series is you know correlating with itself, that is why we this particular term autocorrelation is used to explain this kind of behavior or characteristic of a time series.

Now how do we you know before we can actually account for this autocorrelation in our forecasting model, we need to understand how we can, we need to find out the mechanism where we can compute autocorrelation, so because autocorrelation is also a form of correlation itself so essentially we would be computing correlation between the series and a lagged version of the series itself, so this is how we are going to compute, so you can see the second you know second point and then the first you know sub point computing autocorrelation and correlation between the series and a lagged version of it is going to be used for this.

Now what do we mean by lagged version of the series, so a lagged series with lagged 1, if the series is being lagged by just 1 unit in time, so that would be called series with the lagged 1, so let's see how it is you know defined here, so a lagged series with lagged 1 values of original series moved forward by one time period to occupy one step ahead time point in future, so if this is done what we'll get is a lagged series, series with lagged 1, so original series moved forward by one time period to occupy one step ahead point in future, that means the value that is there at time  $T$ , if you move the series one time period forward then that values would be corresponding to  $T+1$  and the value corresponding to  $T-1$  will now start corresponding for time  $T$ , so therefore the series will create a lagged kind of effect, so this is what we call a lagged series with lagged 1.

So another way to understand the same thing is in effect time index of original series moved back by one time period, so another way we can understand that a particular series the number of riders in you know different months, so if let's say the you know if there are number, particular number of riders in the month of March let's say 2000 and if you know instead of, instead of and before that you know let's say if we had the month of February also, and then January, and then the previous year, so if the ridership data is in that fashion and if we want to you know in a fact if we want to you know move back the time index then you know this particular, the data that is there for the March now if we move back the time index then it would be corresponding to April, right, so the you know the March data would now be the February's ridership data would now be used for the March, so in that sense a lagged effect would be created.

## Regression-Based Forecasting Methods

- Computing autocorrelation
  - Similarly, a lagged series with lag 2
    - Values of original series moved forward by two time periods to occupy two-step-ahead time point in future
      - In effect, time index of original series moved back by two time periods
- lag 1 autocorrelation
  - Measures the linear relationship between values in consecutive time periods
  - Correlation between original series and lag 1 series



Now when we say computing autocorrelations, so once this lagged series is created so we compute the correlation of the original series with this lagged series and that value is taken as the autocorrelation.

Similarly a lagged series with lag 2, so values of original series moved forward by 2 time points to occupy to step ahead time point in future, so once you move the series, once you push the original series in you know forward 2 time points you know, 2 time periods forward in the time so they would be you know corresponding to  $T+2$  and  $T+1$  and you know now the values that you had at you know, the  $T-2$  that would be corresponding to the current time  $T$ , so in that sense the current time  $T$ , the value corresponding value is going to be value that you had at time  $T - 2$  in the original series, so in that sense lagged effect would be created, the same thing we can understand from the time index, if we move the time index of the original series also we can get the same effect.

Now lag 1 autocorrelation, so if we have to define what is lag 1 autocorrelation, so it typically measures the linear relationship between values in consecutive time periods, and so essentially if we look at in terms of how we are going to measure it, correlation between original series and lag 1 series, so we have now understood how we can get the lag 1 series from the original series, now the lag 1 autocorrelation is going to be nothing but the correlation between original series and lag 1 series, and if we want to understand what it means theoretically, theoretically it is, this value is going to measure the linear relationship between values in consecutive time periods, so how the values in consecutive time periods or linearly dependent on each other, so that relationship is going to be you know described by this lag 1 autocorrelation.

```

1 library(xlsx)
2
3 # BicycleRidership.xlsx
4 fulldf=read.xlsx(file.choose(), 1, header = T)
5 fulldf=fulldf[, !apply(is.na(fulldf), 2, all)]
6
7 str(fulldf)
8 tsv=ts(fulldf$Riders, start=c(2004, 1), frequency=12)
9
10 # Compute autocorrelations at lags: 1 to 12
11 # default confidence interval: 95%
12 autolag=acf(tsv, lag.max = 12, main="ACF plot for riders")
13
14 # ACF values
15 data.frame(Lags=12*autolag$lag, ACF=autolag$acf)
16
172 (Top level)

```

Environment History Connections  
 Import Dataset  
 Global Environment  
 Data  
 fulldf 159 obs. of 2 variables

Files Plots Packages Help Viewer  
 Zoom Export

Console Terminal  
 C:/Users/E T Cell/Desktop/MOOC July 2018/Dr. Gaurav Dixit/Session 4/ →  
 Type 'demo()' for some demos, 'help()' for on-line help, or  
 'help.start()' for an HTML browser interface to help.  
 Type 'q()' to quit R.  
 > library(xlsx)  
 Loading required package: rJava  
 Loading required package: xlsxjars  
 > # BicycleRidership.xlsx  
 > fulldf=read.xlsx(file.choose(), 1, header = T)  
 >

So what we are going to do is we'll open R studio and do a few exercises to understand what we have discussed so far in a bit more detail. So let's open R studio, so the same data set is going to be used so before this let's load this library XLSX, now let's import the dataset, so you can see the dataset 159 observations, 2 variables, so we are already familiar with this datasets, let's remove any columns if there are any, let's again have a look at the structure of this particular dataset you can see first column month year, and the second column is the number of riders, let's create a time series object from this, so you can see in the environment section TSV time series this has been created.

```

1 library(xlsx)
2
3 # BicycleRidership.xlsx
4 fulldf=read.xlsx(file.choose(), 1, header = T)
5 fulldf=fulldf[, !apply(is.na(fulldf), 2, all)]
6
7 str(fulldf)
8 tsv=ts(fulldf$Riders, start=c(2004, 1), frequency=12)
9
10 # Compute autocorrelations at lags: 1 to 12
11 # default confidence interval: 95%
12 autolag=acf(tsv, lag.max = 12, main="ACF plot for riders")
13
14 # ACF values
15 data.frame(Lags=12*autolag$lag, ACF=autolag$acf)
16
172 (Top level)

```

Environment History Connections  
 Import Dataset  
 Global Environment  
 Data  
 fulldf 159 obs. of 2 variables  
 values  
 tsv Time-series [1:159] from 200...

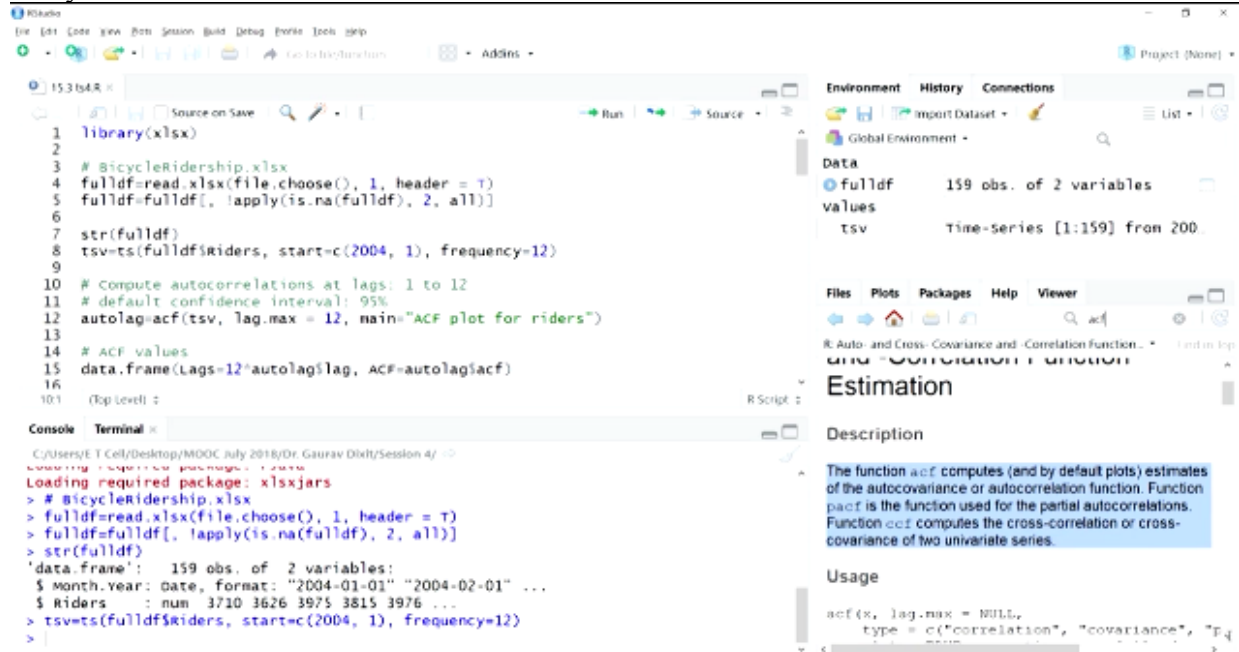
Files Plots Packages Help Viewer  
 Zoom Export

Console Terminal  
 C:/Users/E T Cell/Desktop/MOOC July 2018/Dr. Gaurav Dixit/Session 4/ →  
 Loading required package: rJava  
 Loading required package: xlsxjars  
 > # BicycleRidership.xlsx  
 > fulldf=read.xlsx(file.choose(), 1, header = T)  
 > fulldf=fulldf[, !apply(is.na(fulldf), 2, all)]  
 > str(fulldf)  
 'data.frame': 159 obs. of 2 variables:  
 \$ Month.year: date, format: "2004-01-01" "2004-02-01" ...  
 \$ Riders : num 3710 3626 3973 3815 3976 ...  
 > tsv=ts(fulldf\$Riders, start=c(2004, 1), frequency=12)  
 >

Now you know what we are going to do is we are going to compute autocorrelations at various you know at different lags, so we'll take 1 to 12, so we'll you know compute autocorrelation between original series and lag 1 series and lag 2 original series, lag 2 series original series, lag



3 series original series, lag 4 series and so on up to lag 12, so we are going to compute these many autocorrelations.



Default confidence interval for these computation is going to be 95% which is typically the default in the function that we are going to use, so the function in R that we are going to use to compute this is ACF, if you are interested in finding more detail about this function you can go into the help section and then we can you know we'll just type ACF, and you would see some more detail about this particular function you can see here, the function ACF computes estimates of the auto covariance or autocorrelation function, more detail on the different arguments that are part of this function you can find out from here, right, so we are going to use this function in the first argument you can see we are passing this time series object that we have just created and lag.max is 12 so you can see here lag.max is 12 is the indicating the



```

1 library(xlsx)
2
3 # BicycleRidership.xlsx
4 fulldf=read.xlsx(file.choose(), 1, header = T)
5 fulldf=fulldf[, !apply(is.na(fulldf), 2, all)]
6
7 str(fulldf)
8 tsv=ts(fulldf$riders, start=c(2004, 1), frequency=12)
9
10 # Compute autocorrelations at lags: 1 to 12
11 # default confidence interval: 95%
12 autolag=acf(tsv, lag.max = 12, main="ACF plot for riders")
13
14 # ACF values
15 data.frame(Lags=12*autolag$lag, ACF=autolag$acf)
16
101 (Top level)

```

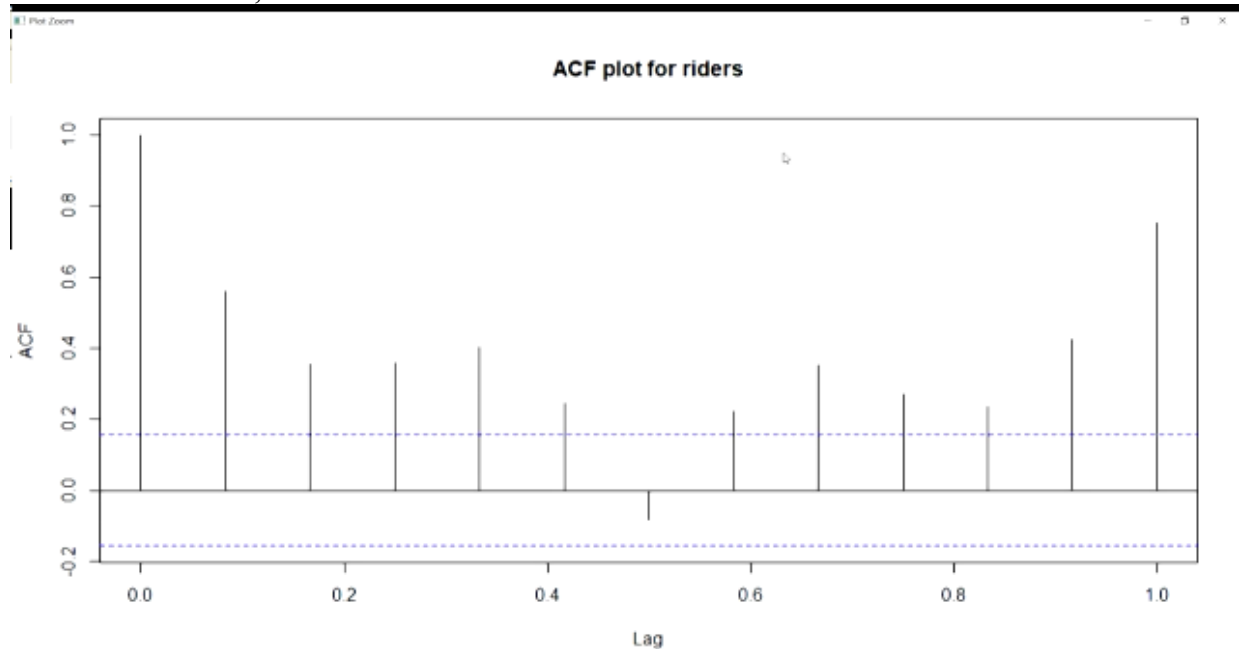
**Environment** History Connections  
 Global Environment  
**Data**  
 fulldf 159 obs. of 2 variables  
 values  
 tsv Time-Series [1:159] from 200

**Files** Plots Packages Help Viewer  
 acf

R: Auto- and Cross-Covariance and -Correlation Function...  
 acf(x, y, lag.max = NULL, type = c("correlat",  
 plot = TRUE, na.action = na.fail, ...)  
 ## S3 method for class 'acf'  
 x[, j]

**Arguments**  
 x, y a univariate or multivariate (not ccf)  
 numeric time series object or a numeric  
 vector or matrix, or an "acf" object.  
 lag.max maximum lag at which to calculate the acf.  
 Default is  $10 \cdot \log_{10}(N/m)$  where  $N$  is the  
 number of observations and  $m$  the number  
 of series. Will be automatically limited to  
 one less than the number of observations in

maximum number of lags for which we want to compute the you know, at which we want to compute these autocorrelations, right, and then this title for the plot, so part of, as part of this function we'll also get a plot so that is also going to be created, so let's run this, we're going to store all this information and the result of ACF function in this because few values that are going to be written from this particular function we might be using it later on, so you can see if you scroll down the help section you would be able to see the values that are going to be written from this functions, so let's run this.

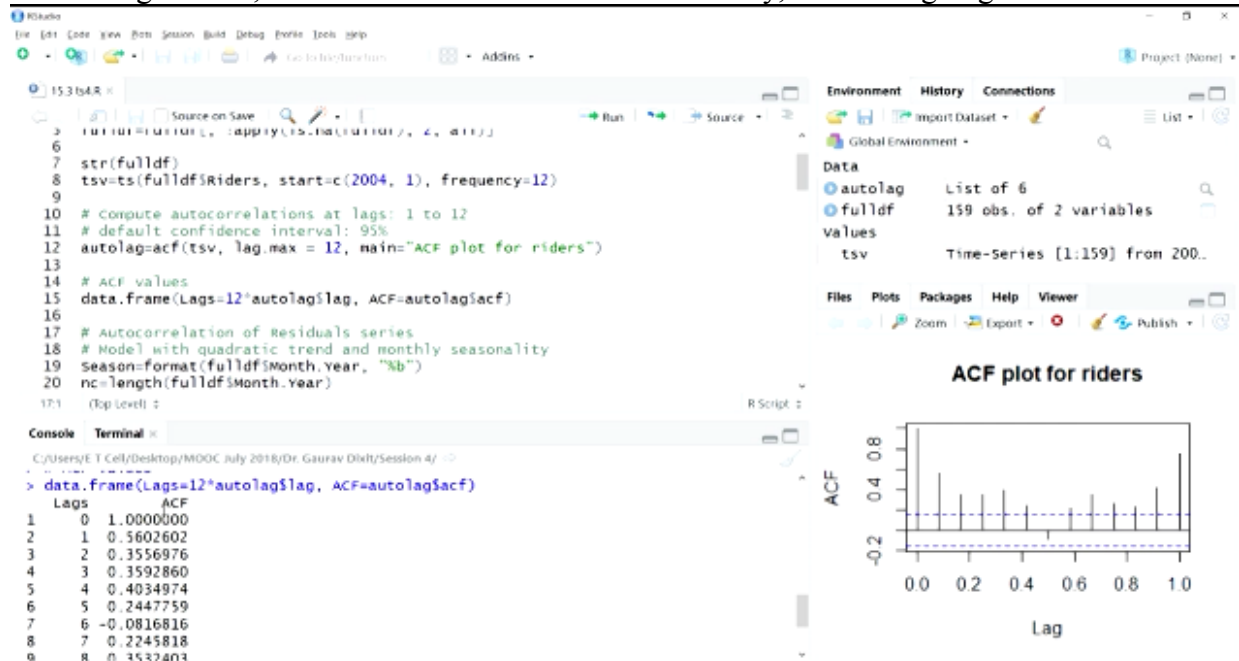


So you can see a plot for riders has been created, now let's you know analyze this plot, so you can see these blue, you know blue dots these are the boundaries you know upper based on the UCI and LCI based on the upper you know confidence level, the upper line and the lower line

based on the confidence level that we are going to use, so because some of these you know we can see these bars, you know single, you know vertical lines that we can see here, so they are representing the, so this line corresponding to the value on X axis of 0, this is corresponding to the original series, then we have the next vertical that we have, this is corresponding to lag 1 series, then the similarly you know lag 2, lag 3 and so on. So in this fashion we can see you know number of you know lags and autocorrelation between the original series and this lag series, lagged series have been computed.

So if we look at the you know this output, most of these lags starting from lag 1 to lag 2 and lag 3, we can see most of them are going outside these specified you know boundary lines based on the confidence interval, so this in a way is indicating high autocorrelation between original series and the lagged series, so if we look at the correlation between original series and lag 1 series, so this particular bar, this particular vertical line and its corresponding values indicating the level of autocorrelation, so it is near about 0.6, less than 0.6, so it seems to be highly correlated.

Similarly for other lags also we can see most of these lines they are on the higher side, so therefore the lagged series are highly correlated with the actual original series, and therefore it becomes important for our you know forecasting models to incorporate this information, because this is going to be you know significantly improving the performance of our forecasting models, so let's look at these values numerically, so we are going to create this data



frame to have a look at this numbers, so you can see here, for different lags so lag 0, so which is the original series that you know autocorrelation of the original series with the original series itself so that is off course going to be 1 so that we can see. Then for lag 1 it is 0.56 and then for lag 2 it is about 0.36, so in this fashion we can see most of these lagged series are highly correlated with the, with the you know original series.



---

## Regression-Based Forecasting Methods

- Open RStudio
- Directions of autocorrelation behaviors
  - Positive autocorrelation
    - High values are followed by high values
    - Low values are followed by low values
  - Negative autocorrelation
    - High values are followed by low values
    - Low values are followed by high values

let's understand a few more aspects related to autocorrelation behaviors, so one of these things is direction of autocorrelation behaviors, so as we have seen in the output itself that the autocorrelation values just like the correlation values they could be positive or negative, in a sense they indicate the you know direction of this correlation behavior, so here in this case also positive autocorrelation it means high values are followed by high values, you know, low values are followed by low values, so whenever we get a number and if it is in the positive side whether this is applicable for correlation values also and autocorrelation values also, so high values if the value is positive this autocorrelation value is positive then that means high values are followed by high values, and low values are followed by low values, so if we get the negative autocorrelation value so that indicates that high values are typically followed by low values and low values are typically followed by high values, so this direction of autocorrelation behavior is important for us to understand that this is something that is going to help us in terms of identifying different time series component and later on incorporating them in our forecasting model, so let's move further.

## Regression-Based Forecasting Methods

- Which autocorrelation behaviors should be explored?
  - Stickiness
    - Strong and positive lag 1 autocorrelation indicates this behavior in the series
      - Consecutive values move in the same direction
  - Swings
    - Strong and negative lag 1 autocorrelation indicates this behavior in the series
      - Alternate values move in the opposite direction
  - Cyclical pattern
    - Strong positive or negative autocorrelation at lag  $cl$ 
      - Where  $cl$  is cycle length
      - For example, this behavior at lag 12 would mean annual seasonality



So which kind of autocorrelation behaviors should be explored, so one thing is we looked at the ACF plot that in the R studio and we need to analyze that ACF plot to understand the behavior of time series, then it is important for us to understand the you know kind of behavior that we should identify and locate and the kind of behaviors that we can easily incorporate in our forecasting models, so let's discuss some of these behaviors which can be explored, so first one that we are going to discuss is stickiness, so what we mean by stickiness? So it essentially means that strong and positive lag 1 autocorrelation if that is there in the ACF plots as we saw in the R studio, if there is a strong and positive lag 1 autocorrelation you know then this particular thing indicates this kind of behavior stickiness, the strong and positive lag 1 autocorrelation indicates the stickiness in the time series, so what it means is consecutive values move in the same direction, so as we saw in the ACF plot lag 1 had you know high autocorrelation value about 0.56, so that indicates that there is some stickiness in the time series that we are analyzing, so the higher values, the consecutive values they move, they tend to move in the same direction.

Let's look at the another autocorrelation behavior which is swings, so when do we observe this swings kind of behavior in the time series, when a strong and negative lag 1 autocorrelation is present then probably that is going to indicate swings in the series, so what does this mean? Alternate values move in the opposite directions, so therefore you know the first the higher values going to be there then it would be, it is typically going to be followed by low value then it is going to be a typically followed by high value and then again, so high low, high low, so those kind of swings that we see in a particular you know time series that is the kind of behavior, that is the kind of characteristic that is going to be shown if there is a strong and negative lag 1 autocorrelation.

So in the dataset that we are using the time series that we are analyzing, so we had positive you know lag 1 autocorrelation, so probably stickiness is the thing that is the behavior that is being exhibited by this you know riders dataset and not the this kind of behavior swing. Third one is the cyclical pattern, so when this kind of behavior is indicated so strong positive or negative autocorrelation at lag  $CL$ , where  $CL$  is the cycle length, so if we expect that a kind of cycle in

the time series and we see a strong positive or negative autocorrelation at that particular lag, and the lag which is indicating the cycle, in the series so that would in a way indicate towards the cyclical pattern that is there, so for example this behavior at lag 12 would mean annual seasonality, so we looked at the you know when we thought the ACF plot you know we saw that the highest autocorrelation value that was there it was at lag 12, so it means then that probably there is a cyclical pattern so this, that is the lag 12 so this is annual seasonality that seem to be present in the you know riders data that we had.

So as we have been discussing that the regression based forecasting methods that we have been using till now in few lectures, so we were able to model the trend and seasonality and then we had the residual series, so in that residual series that we had in the previous lecture you know we can check autocorrelation that residual series and we can examine you know if there are any patterns you know autocorrelation specific behaviors that are there and then we can model that also in our regression based models. So to examine the adequate modeling of different patterns we can check the residual series, if some pattern has not been model then it would be reflected in the residual series as we have been doing in previous lectures also, for example seasonal pattern so there is going to be no autocorrelation at season lag, so if we are able to adequately model the seasonality component in the time series then autocorrelations would not be visible in the ACF plot at the seasons lag, right, so this is one.

## Regression-Based Forecasting Methods

- Checking autocorrelations of residual series
  - To examine the adequate modeling of different patterns
    - Seasonal pattern: No autocorrelation at season's lag
- Using autocorrelation to improve forecasting models
  - Directly account for autocorrelation into the regression model
    - Such models are called **autoregressive models**
  - Multi-level forecasting
    - Second level forecasting model on residual series

So using autocorrelation to improve forecasting model, so one thing that can be done is we can directly account for autocorrelation into the regression model itself, so such models are called autoregressive models, so for example if we know that our original series is going to be depending you know highly depending on the previous you know points ridership, so if at time T the riders number of riders are there and this number is highly dependent on the number of riders that we had in the previous month, so this particular aspect can be you know included in the regression model itself, and when this kind of thing is done we call this models as autoregressive models, so this is one approach to incorporate autocorrelation and improve the forecasting performance.



In the second approach is multilevel forecasting, so in this case what we do is you know first we apply our you know regression based models and the residual series then again used to build a second level forecasting model, so different levels are used to adequately you know capture different aspects of time series, so autocorrelation could be captured in the some you know second level of you know the modeling.

## Regression-Based Forecasting Methods

- Autoregressive (AR) models
  - Linear regression models where
    - Predictors are the past values of the same series
  - Linear regression model with original series as output variable and lagged series as predictor
- More general class of such models is called autoregressive integrated moving-average (ARIMA) models

Now let's discuss a bit about autoregressive models or AR models, so these are linear regression models where predictors are the past values of the same series, right, so as we understood that the values at time T if they are dependent on the values at time T-1 and other you know lagged you know points then this can also be set that predictors or the past values of the same series.

Linear regression model with original series as output variable and lagged series as predictor or the autoregressive models, so the linear regression model typically typical standard multiple linear regression equation is  $Y = \alpha + \beta_0 X_1 + \beta_1 X_2 + \beta_2 X_3$ , now in this instead of  $X_1$  we can have  $Y_{T-1}$  so what  $Y_{T-1}$  is indicating is the lagged series, and that lagged series is being used as a predictor in the, in a way in terms of predicting the values at time T, so  $Y_T$  is going to be  $\alpha + \beta_0$  for costing term, plus  $+ \beta_1 Y_{T-1} + \beta_2 X_2 + \beta_3 X_3$  and so on, so in that sense the lagged series can also be used as a predictor in the regression model itself.

Now more general class of such model is called autoregressive integrated moving average or ARIMA models, so these AR models they are extended and the moving average aspect was that is also integrated then we get, we reached to the more general class which is called ARIMA models.



## Regression-Based Forecasting Methods

- Autoregressive (AR) models
  - For example, an AR2 (autoregressive model of order 2) can be expressed as:
$$Y_t = \beta_0 + \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \epsilon$$
- Estimation of the regression model with AR terms
  - ARIMA estimation should be preferred over ordinary linear regression estimation
    - Because of its ability to account for the dependency between observations



Now let's look at the you know one example, let's look at one example, so for example AR2 if we are looking for autoregressive model of order 2 then how the equation regression equation can be written as you can see here  $Y_t = \beta_0 + \beta_1 Y_{t-1} + \beta_2 Y_{t-2}$ , so you can see this autoregressive model is of order 2, so that essentially means we have 2 autoregressive terms here in the regression equation, so one is  $Y_{t-1}$  and the another one is  $Y_{t-2}$  and that is why the order of this auto regressive model is indicated as 2.

So then the next important aspect of the autoregressive models is estimation, so estimation of the regression model with AR terms so one approach is that ARIMA estimation and techniques that could be used and there are certain reasons for it to be preferred over ordinary linear regression estimation or OLS estimation that reason is also something that we have discussed before, because of its ability to account for the dependency between observations, so if we are going to use OLS regression the estimates that we are going to be computing they would actually be you know also based on the assumption that observations are you know cases they are independent, but if we use the ARIMA estimation method they account for this dependency and therefore the estimates can be more accurate.

Now in the AR modeling if we look at from the modeling prospective, how do we determine the order of the model, because a series could have high correlation with a number of you know lagged series of itself, so therefore how do we determine the order of the series, where do we stop or where what order do we finalize, so choosing the order of the model is a tough task, you know specially in the context when you know original series contains trend and seasonality patterns, so when both these patterns are present that it becomes even more you know complex, so what can be done is several initial data transformation can be performed and these data transformation could essentially you know indicate us in a sense the order of the model that could be used.

## Regression-Based Forecasting Methods

- AR modeling
  - Choosing the order of the model is a tough task
    - Specially when original series contains trend and seasonality patterns
      - Several initial data transformations would have to be performed
- Instead of AR models, the more general ARIMA models can also be used
  - Require higher level of statistical expertise
  - Considered less robust

So another approach of AR modeling could be that instead of AR models we can also use the more general ARIMA models, so why not use, why just stop at you know AR level, why not use ARIMA models, so however these ARIMA models they have their own pros and cons, so few important aspect about these ARIMA models is that they require higher level or statistical expertise, the kind of you know data transformation the kind of statistical expertise that we would be require, that would be you know more engaging, and also they are considered less robust, so every time probably you you know apply an ARIMA model and you do certain changes in your modeling then the results might varies, so the robustness is you know that is under question, that is why ARIMA models are not that popular in the you know forecasting you know in the forecasting arena.

## Regression-Based Forecasting Methods

- Use of multi-level forecasting
  - To incorporate autocorrelation
  - Simpler and straightforward approach
  - Requires a second-level forecasting model for the residual series
    - Short-term forecasting performance might improve due to utilization of autocorrelation



Now let's discuss the another aspect use of multilevel forecasting something that we discussed also, so main idea is you know to incorporate autocorrelation, so this particular approach is instead of using ARIMA which is less robust and requires more statistical expertise we can use multilevel forecasting where you know to incorporate autocorrelation, simple state forward approach so what we need to do is you know it will require us to build a second level forecasting model for the residual series, so we'll have a first level forecasting model which would be you know, which would be capturing the you know different components of the time series, trend, seasonality, and the residual series of you know that model then we can look for the you know incorporating autocorrelation, so we can develop a second level forecasting model, so short term forecasting performance might improve to do due to utilization of autocorrelation, why we call it short term? Because if we say that you know autocorrelation essentially means that neighboring values are dependent on each other, so therefore short term forecasting would actually be you know influenced by this dependence, and therefore short term forecasting performance if we incorporate autocorrelation then this short term performance can be improved, so the second level forecasting model is essentially going to help us in improving the short term performance, because the long term performance they might be determined by trained component and the seasonality component which we have already, which we should have already you know adequately model in the first level forecasting, so second level we can focus on, we can incorporate autocorrelation and improve the short term forecasting.

## Regression-Based Forecasting Methods

- Steps:
  - Use first-level method (e.g., multiple linear regression) to forecast future values of the original series
    - To capture trend and seasonality patterns
    - Example, quadratic trend and seasonality modeling for Bicycle Ridership dataset
  - Use second-level method (e.g., AR model) to forecast residual series
    - Need for initial data transformations is not required since residual series is not expected to have any trend or seasonality patterns
  - Combine the results to produce the final forecast
    - Final forecast = First-level forecast + Second-level forecast



So what are the steps? So we use first level method multiple linear regression to forecast future values of the original series, main idea is to capture trend and you know seasonality patterns, so example something that we have done in previous lectures quadratic trend and seasonality modeling for bicycle ridership dataset that we have done, so this is one that we do in first level method and the second level method that we are specifically discussing in this lecture the AR model can be used to forecast residual series and need for initial data transformation is not required, since residual series is not expected to have any trend or seasonality patterns, so these trend and seasonality pattern we should have been able to adequately capturing the first level and the second level therefore only the AR components would be remaining, and we would be modeling them in the second level method.

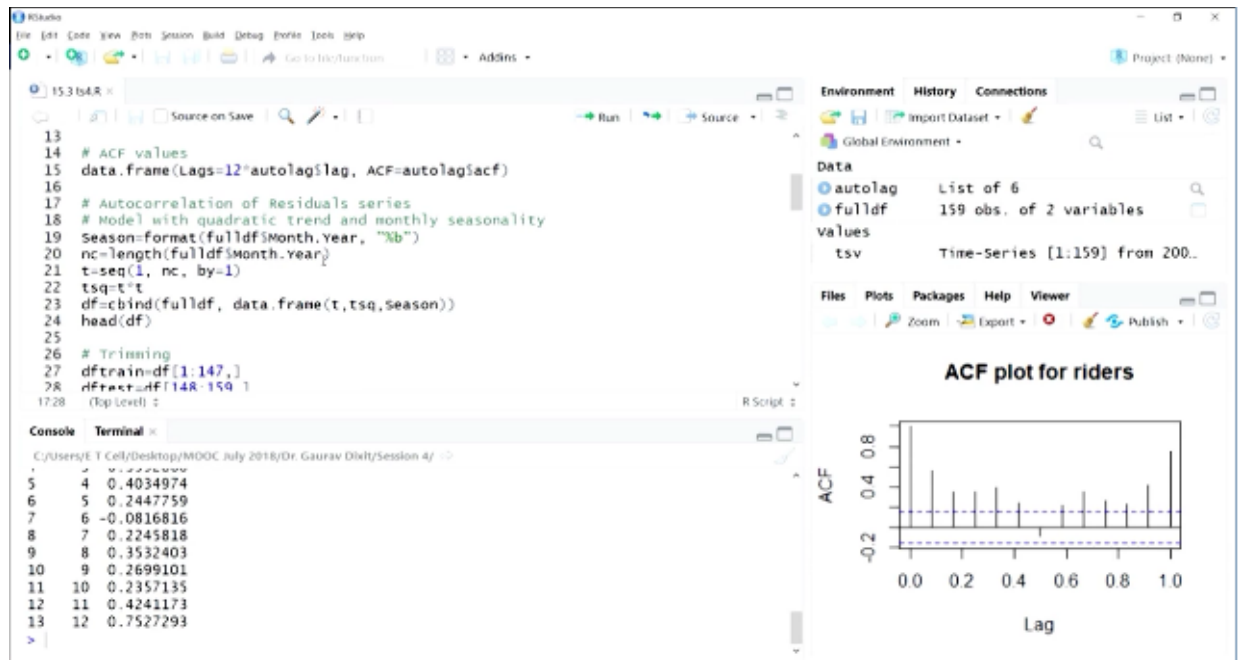
## Regression-Based Forecasting Methods

- Fitting AR model to the residual series
  - Examine the autocorrelations of the residual series
  - Choose the order of the AR model
    - Based on the lags where autocorrelation appears
  - AR1 model for the residual series, where  $R_t$  denotes the residual at time  $t$

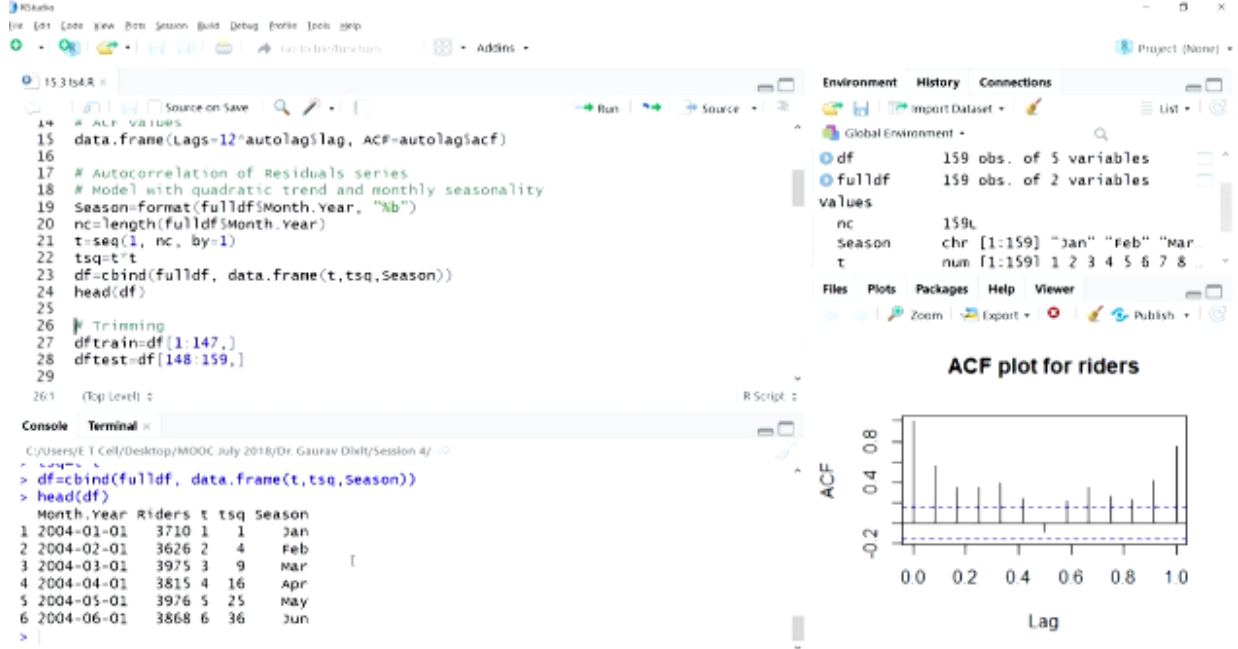
$$R_t = \beta_0 + \beta_1 R_{t-1} + \epsilon$$

- Open RStudio

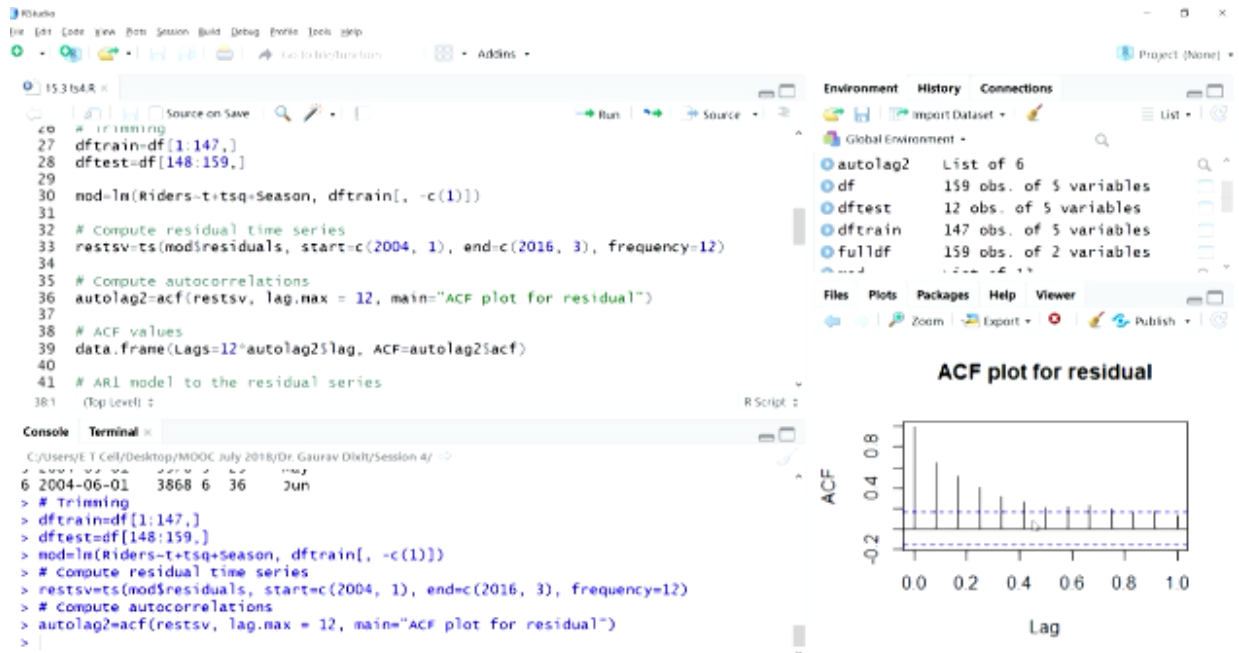
If we talk about the ARIMA you know this transformation, data transformation would be required because everything is typically done in the same go, in the same you know in one model itself, so combine the results to produce the final forecast, so final forecast is going to be first level forecast plus second level forecast. Now fitting AR model to the residual series, so first you know we can examine the autocorrelations of the residual series we can then select the order of the AR model based on the lags you know where autocorrelation appears, so AR 1 model for the residual series you know if we are denoting you know  $R_t$  is the residual series then we can write in this fashion, so for the residual series  $R_t$   $\beta_0 + \beta_1 R_{t-1}$ , so in this fashion we can write the AR 1 model. Similarly other you know models can also be written in the similar fashion, so what we are going to do is we'll open R studio and then we will do an exercise to understand a bit more about the auto regression, so let's see here that we are just quickly, we'll just quickly run through this part because we have done this in previous lecture.



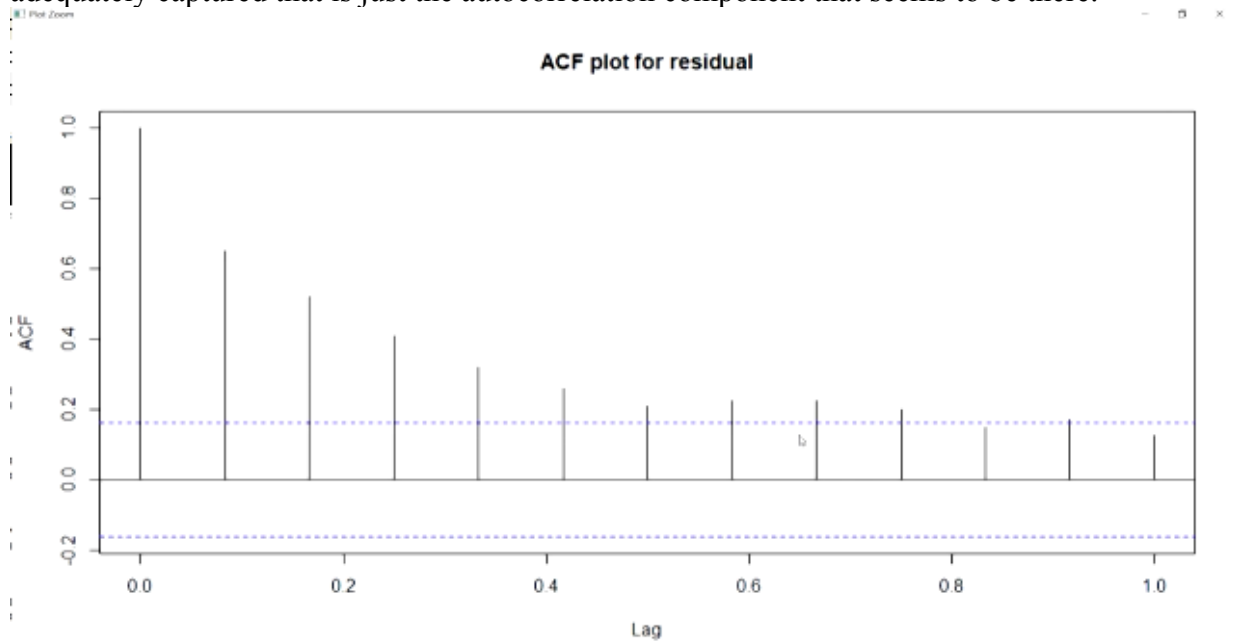
Let's compute season, this length because we would be computing T, T square, and then let's compute this data frame, let's look at the first 6 values you can see riders T, T square, and



season, so this quadratic model we had you know done in previous lecture also, let's do the trimming, so DF train and DF test, now we are going to just build this linear regression model using LM function, so this will give us this residuals, so using this residuals values we can create our residual time series so let's do this.



Now we are interested in understanding this particular residual time series, so again you can see we are using ACF function here to look at the autocorrelation, we have the lag max 12, so let's compute this, so you can see here in the plot section that we can see high autocorrelation values for lag 1, 2, 3 onwards, if you are interested in the actual numerical values we can see from here as well that you know 0.65152 so we can see in this fashion, now you would see at lag 12 the value you know autocorrelation value is between the bounce here, and also the numerical value is also on the lower side, so that we can see that seasonality and those aspect have been adequately captured that is just the autocorrelation component that seems to be there.



So what we can do is we can build an AR1 model on this residual series, so for this we can use AR function which is available, so in this AR function more details you can find out in the help section, can see we are you know passing on this residual time series or a max is 1, because we



are building any R1 model, why we are building AR1 model is because we feel that once you capture AR1 this relationship will propagate itself and the other you know orders we don't need to explicitly include in the equation, because once we do AR1 if this relationship is going to propagate and more often they're not this typically gives us a good enough model, so let's run this.

The screenshot shows the RStudio environment with the following components:

- Source Editor:** Contains R code for computing residuals, autocorrelations, and fitting AR1 and ARIMA models.
 

```

32 # Compute residual time series
33 restsv=ts(mod$residuals, start=c(2004, 1), end=c(2016, 3), frequency=12)
34
35 # Compute autocorrelations
36 autolag2=acf(restsv, lag.max = 12, main="ACF plot for residual")
37
38 # ACF values
39 data.frame(Lags=12*autolag2$lag, ACF=autolag2$acf)
40
41 # AR1 model to the residual series
42 mod1=ar(restsv, order.max = 1, method = "ols")
43 mod1
44 # Using ARIMA model for AR1
45 mod2=arima(restsv, order = c(1,0,0))
46 mod2
47
481 (Top Level)
      
```
- Environment:** Lists objects in the workspace:
  - dfctest: 12 obs. of 5 variables
  - dftrain: 147 obs. of 5 variables
  - fulldf: 159 obs. of 2 variables
  - mod: List of 13
  - mod1: List of 15
- Console:** Shows the output of the `ar()` function:
 

```

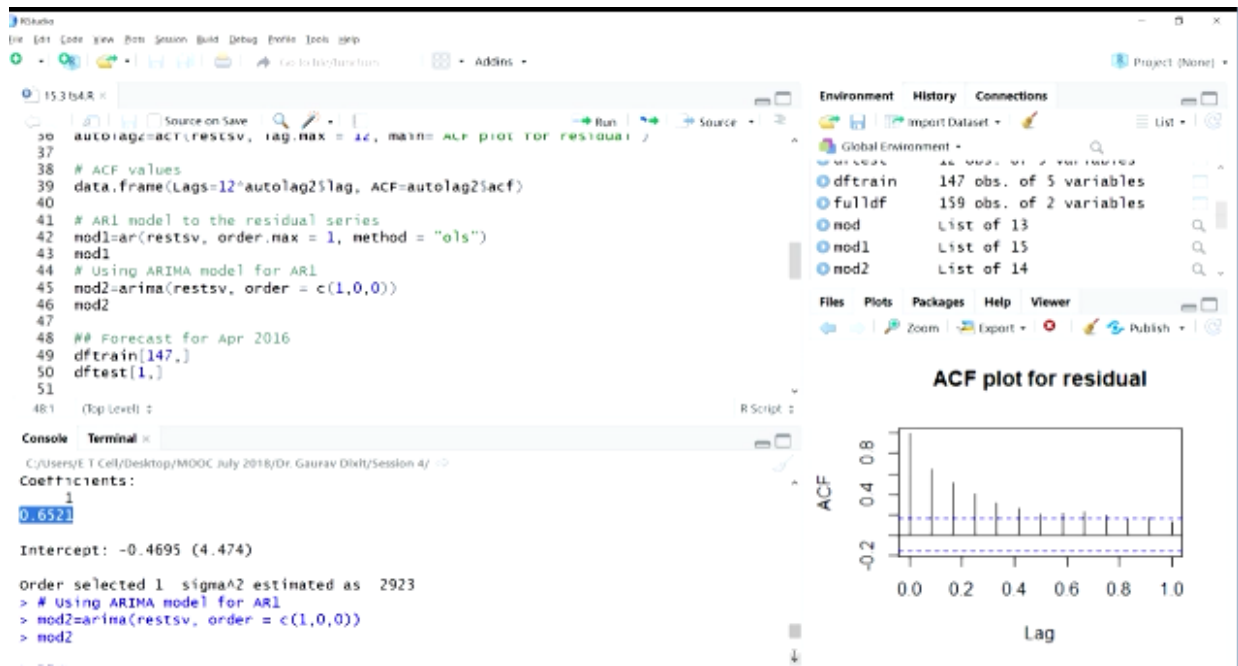
C:/Users/E T Cell/Desktop/MOOC July 2018/Dr. Gaurav Dixit/Session 4/ >
ar(x = restsv, order.max = 1, method = "ols")

coefficients:
 1
0.6521

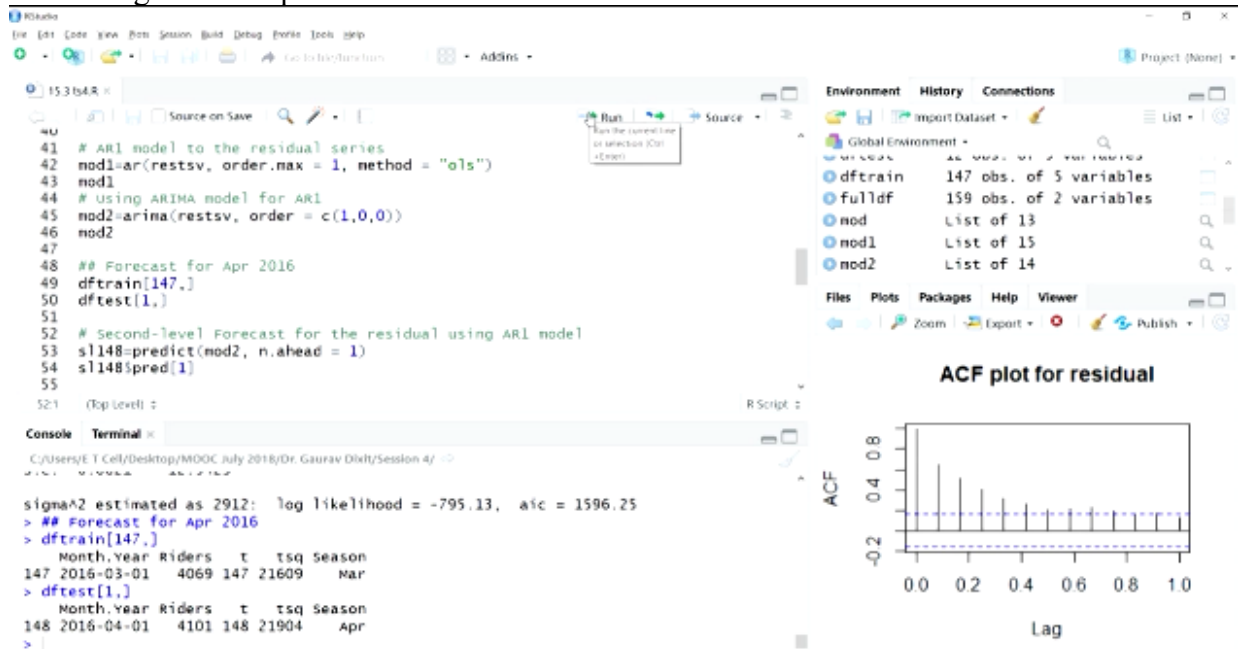
Intercept: -0.4695 (4.474)

order selected 1  sigma^2 estimated as 2923
      
```
- ACF plot for residual:** A bar chart showing the Autocorrelation Function (ACF) for lags 0 to 1.0. The y-axis ranges from -0.2 to 0.8. The plot shows a significant positive autocorrelation at lag 1, which then decays towards zero for subsequent lags.

And let's look at the coefficient here you can see 0.6521, now the same AR1 model we can also will using the ARIMA function, so this is also available, so what we need to do there is we need to specify the order, so since we are just focusing on the AR model, so just you know that order is specified and the others are you know initialize as 0, so we can use this function, more detail on this ARIMA function as well you can find the help section, again if we look at the results, now if you look at the coefficient value this comes out to be 0.6496, if you go back and the

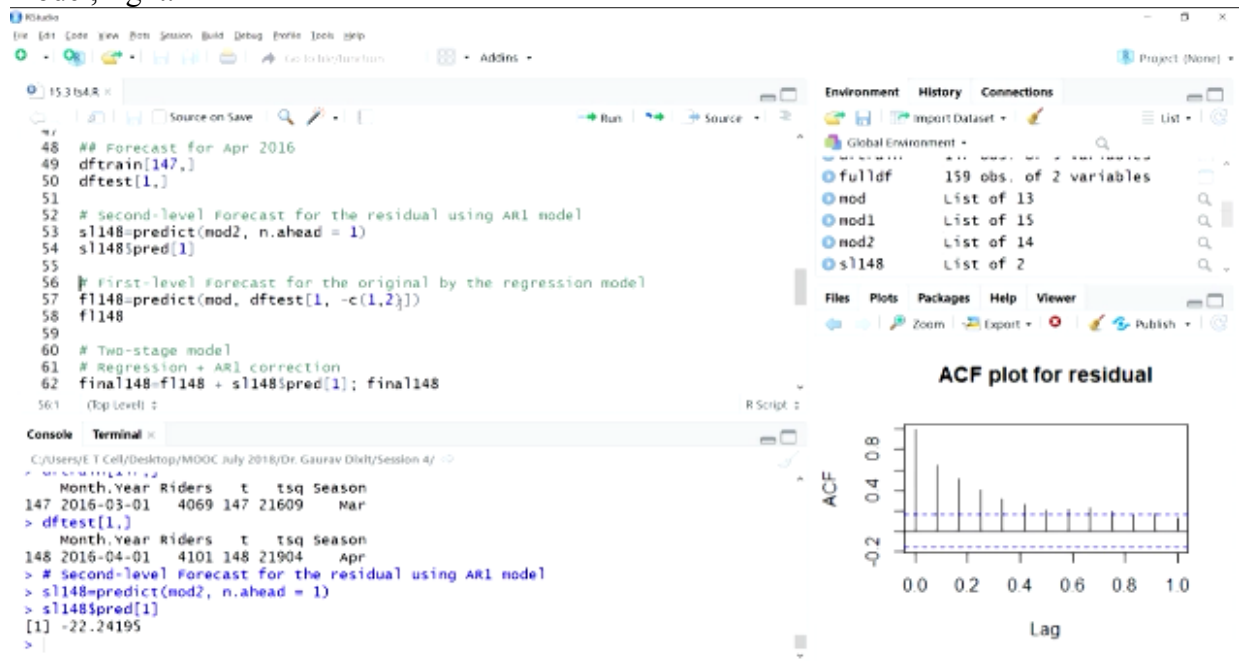


coefficient value that we had using the OLS it was 0.6521, so there is slight difference, slight adjustment in the ARIMA estimation method in comparison to OLS, so OLS because it is assuming the observations or independent, so 0.6521 is there, the coefficient is there for AR1 model, but if we look at the ARIMA estimation so this value decreases a bit, it becomes 0.6496, so that is the you know difference in the estimation because the ARIMA estimation is accounting for the dependence between observations as well.

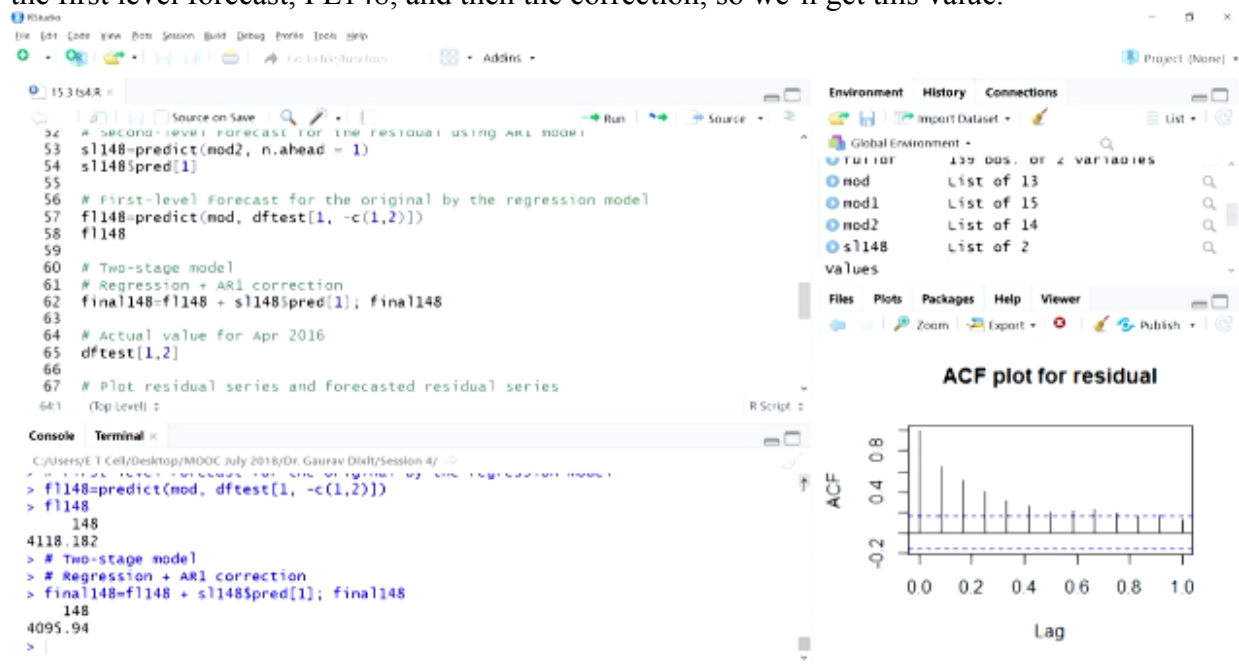


Now we can you know have a look at the you know forecast for the you know we'll look at the April 2016 this values, till March 2016 we have used for the training, so we can look at the actual value that was there 4069, DF test so we can have a look at this value, so this is the value so now if we are applying, if we use the predict function to use this model and make forecast for the 148 value that is part of the test partition, the first value of test partition, so this is the

actual value and we are trying to predict this, so let's run this code, and let's have a look at the value, so this is the value, this is the forecast for the residual so you know this again can be used, so this is the second level forecast that we have now just done for the residual using AR1 model, right.



Now if we look at the first level forecast you know for the, you know original by the regression model so that can also be done, so mod this we have already you know model we have already build can use this, so this was the you know first level forecast 4118 and the second level forecast is -22, so we can you know do a summation of this two forecast and we'll get the our you know two stage models, so first regression, the second one AR1, so you can see final 148 is the first level forecast, FL148, and then the correction, so we'll get this value.



So you can see 4095, so if we compare this with the actual value, actual value is here, you can see actual value is 4101 so after the second level forecast the final value that we get this value is much closer to this actual value in comparison to the first level forecast that we had, right, so the second level forecast, the autocorrelation you know when we incorporated autocorrelation that is present in the series, our forecasting performance actually improved, so that is very clearly visible in this values as well.

The screenshot shows the RStudio interface with the following components:

- Source Editor:** Contains R code for a two-stage model:
 

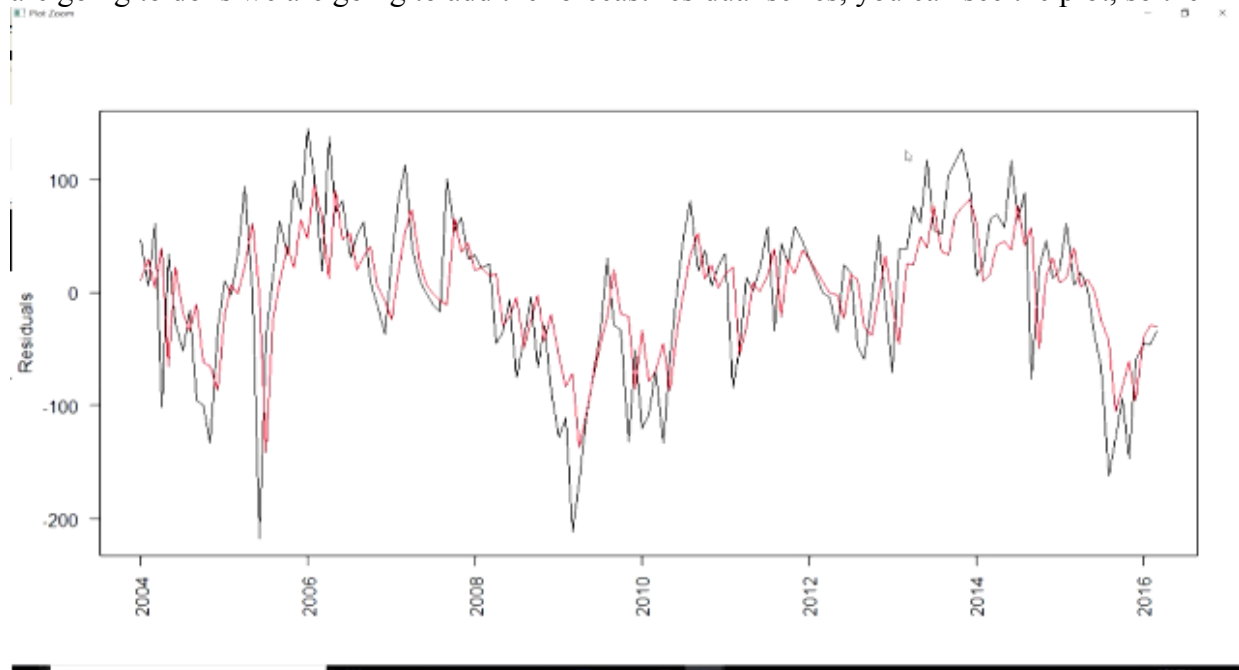
```

60 # Two-stage model
61 # Regression + AR1 correction
62 final148=f1148 + s1148$pred[1]; final148
63
64 # Actual value for Apr 2016
65 dftest[1,2]
66
67 # Plot residual series and forecasted residual series
68 plot(restsv, xlab="", ylab="Residuals", las=2)
69 tindex=time(restsv)
70 points(tindex, restsv-mod2$residuals, type="l", col="red")
71
72 # ACF values for residuals of residual series
73 autolag3=acf(mod2$residuals, lag.max = 12)
      
```
- Console:** Shows the execution output:
 

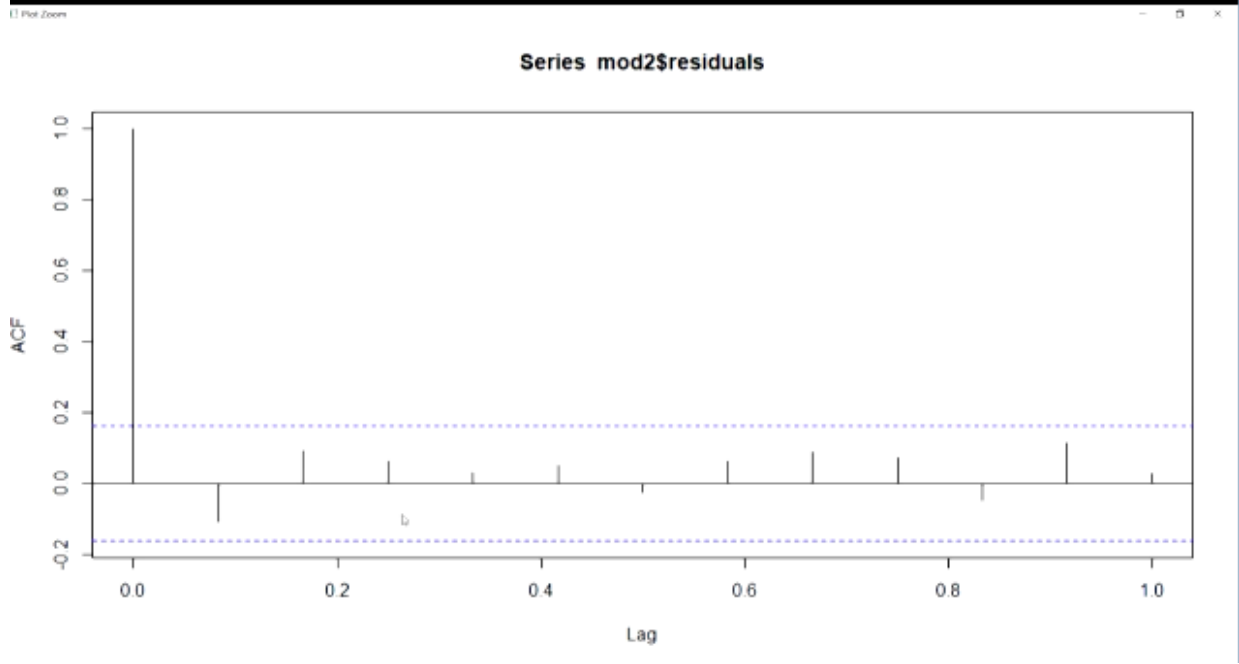
```

4118.182
> # Two-stage model
> # Regression + AR1 correction
> final148=f1148 + s1148$pred[1]; final148
148
4095.94
> # Actual value for Apr 2016
> dftest[1,2]
[1] 4101
      
```
- Environment:** Lists objects: mod (List of 13), mod1 (List of 15), mod2 (List of 14), s1148 (List of 2).
- ACF plot for residual:** A plot showing Autocorrelation Function (ACF) values for lags 0 to 10. The y-axis ranges from -0.2 to 0.8. The plot shows a significant peak at lag 1, indicating autocorrelation in the residuals.

Now we can plot this residual series and forecast residual series to see how adequately it is been model in overall sense, so let's plot, you can see this is the plot for the residuals, now what we are going to do is we are going to add the forecast residual series, you can see the plot, so the



red line this is quite you know closely following this particular you know plot, so we can see that you know that is why this is adequately being modeled, AR component is adequately being modeled, we can look at the ACF values also to again reconfirm the same thing, so we can call ACF function on this and we can see now here in the ACF plot as you can see, if we zoom in no



lagged you know autocorrelation values had different lag is beyond the bound that blue line that we see, so therefore the autocorrelation has been adequately modeled here.

The screenshot shows an RStudio session with the following components:

- Environment:** Lists objects including `autolag3` (List of 6), `df` (159 obs. of 5 variables), `dfctest` (12 obs. of 5 variables), `dftrain` (147 obs. of 5 variables), and `fulldf` (159 obs. of 2 variables).
- Console:** Shows the execution of `data.frame(Lags=12*autolag3$lag, ACF=autolag3$acf)` resulting in the following table:

Lags	ACF	
1	0	1.0000000
2	1	-0.10802954
3	2	0.09274124
4	3	0.06303465
5	4	0.03082994
6	5	0.04965573
7	6	-0.02507292
8	7	0.06209714
9	8	0.08934047

So if you are interested in the numerical values you can see here most of the values they are you know quite low, so we can see here, so let's go back to our discussion.

## Regression-Based Forecasting Methods

- Evaluating predictability of a time series
  - Whether applying forecasting methods on a particular time series would yield any meaningful forecasting model
  - Tested by examining whether the series is a **random walk**
    - Random walk is a series where values change from one period to next in a random fashion
      - Example: predicting stock prices



Now another important aspect of regression based forecasting method is evaluating predictability of a time series, so what do we mean by this, whether so essentially what we mean is whether applying forecasting methods on a particular time series would yield any meaningful forecasting model, so for any time this is specifically true for a time series that before we attempt to build a forecasting model for a particular time series we need to make sure whether that series has some forecasting value for us or not, so how do we find out? Otherwise different models and different things that we might try out, they might not work well or might not provide us safe you know meaningful forecasting you know performance, so how do we you know understand whether a time series should be forecasted at all, you know that is something that we are going to discuss, so how predictability of a time series can be determined, so this can be tested by examining whether the series is a random walk, right, so how do we do this? So let's first understand what is a random walk, so random walk is a series where values change from 1 period to next in a random fashion, so for example predicting a stock prices, so typically it is understood that from the fusion market hypothesis that this asset prices they are you know, they are randomly you know because it is very difficult to determine how you know, how they are, how they are computed, how they are determined, so predicting a stock prices is kind of a random walk and no information is going to help us in terms of building a model and having a meaningful forecast, so if we are able to test for a series whether the values change from one period to next in a random fashion, then we would have a sense whether there is you know any, you know, any purpose in terms of predicting a time series.

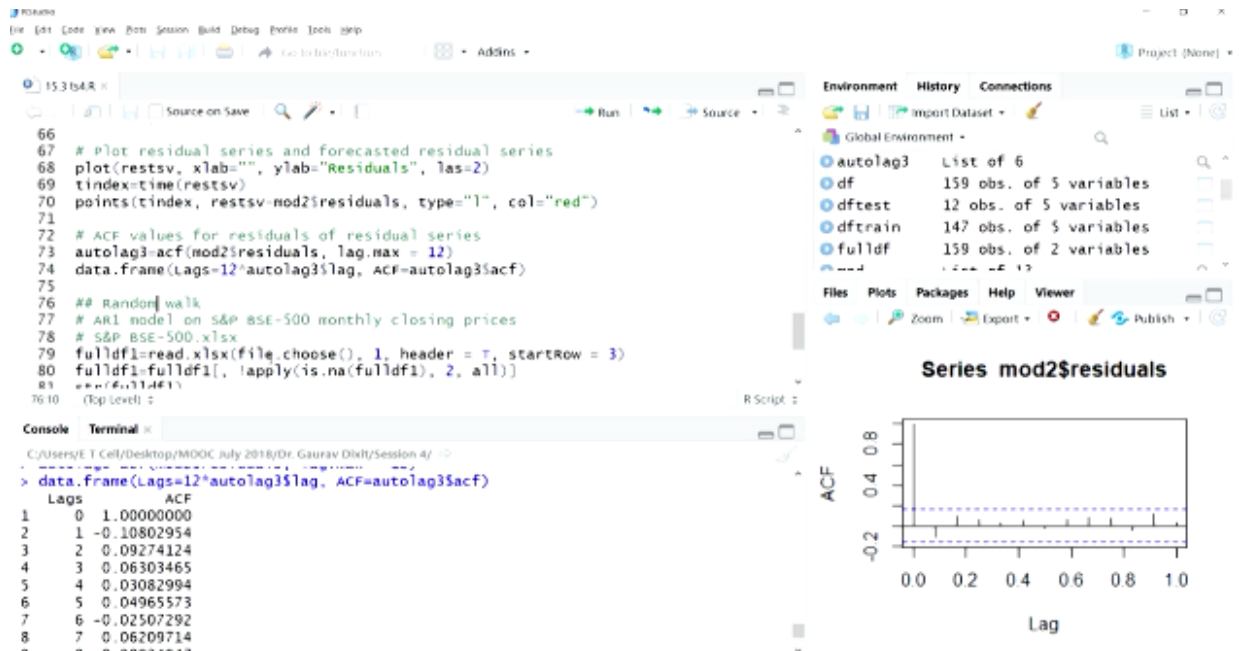
## Regression-Based Forecasting Methods

- Random walk
  - Special case of an AR1 model
    - Where  $\beta_1 = 1$ 
$$Y_t = \beta_0 + Y_{t-1} + \epsilon$$
  - To test whether a series is a random walk
    - AR1 model is fitted
    - $H_1: \beta_1 \neq 1$  ( $H_0: \beta_1 = 1$ )
- Open RStudio

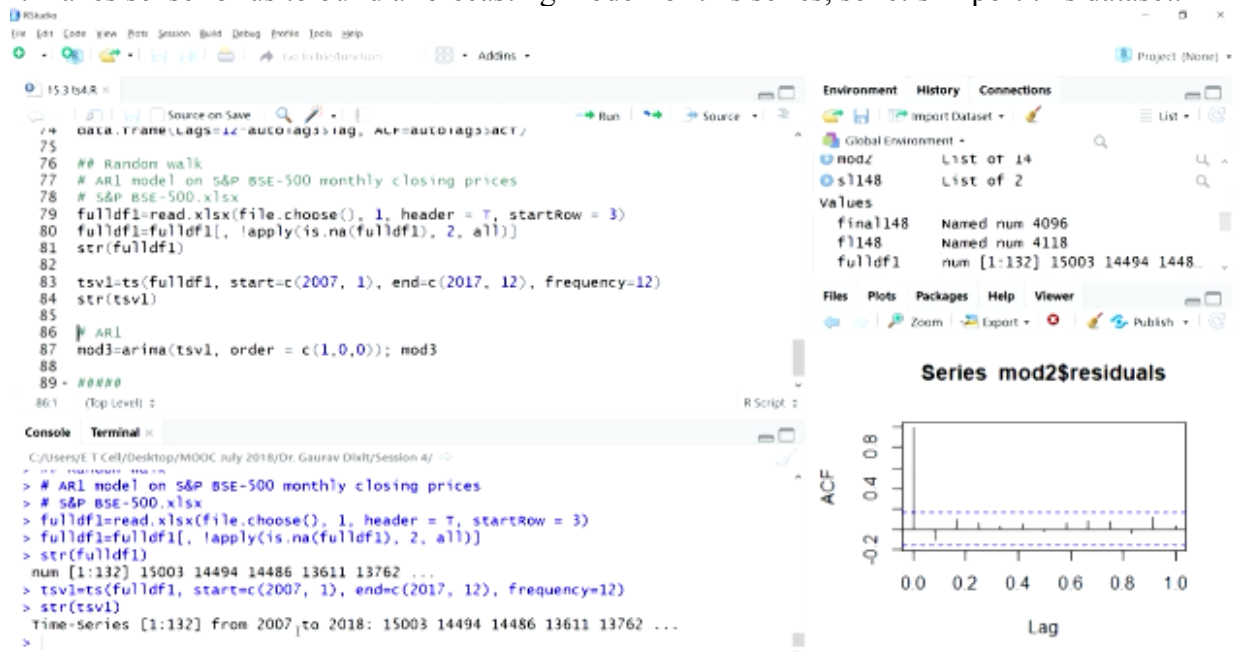
So how do we define random walk model? So this is a special case of an AR1 model, where beta 1 is 1, so the equation can be written in this fashion  $Y_t = \beta_0 + Y_{t-1} + \epsilon$ , so you can see the coefficient beta 1 is 1 here, so that means  $Y_t$  is you know dependent on the previous value that is  $Y_{t-1}$  and then the noise term, so that means if anything is part of noise so and if this most and this you know the value is mainly dependent on  $Y_{t-1}$  and the remaining part is noise which is something that we cannot you know further you know use for improving the forecast, so if this is the case then probably you know the series is a random walk.

Now how do we test? So we can rewrite this whole thing as to test whether a series is random walk, AR1 model is fitted, so that is first one, so essentially we are checking for this hypothesis, this alternate hypothesis whether beta 1 is, whether beta 1 is equal to 1 or not, so beta 1 not equal to 1, if beta 1 = 1 then null hypothesis, it is null hypothesis that is going to be accepted that means the series is a random walk, if it not accepted then probably there is some scope, some you know, some forecasting of that series can be done.



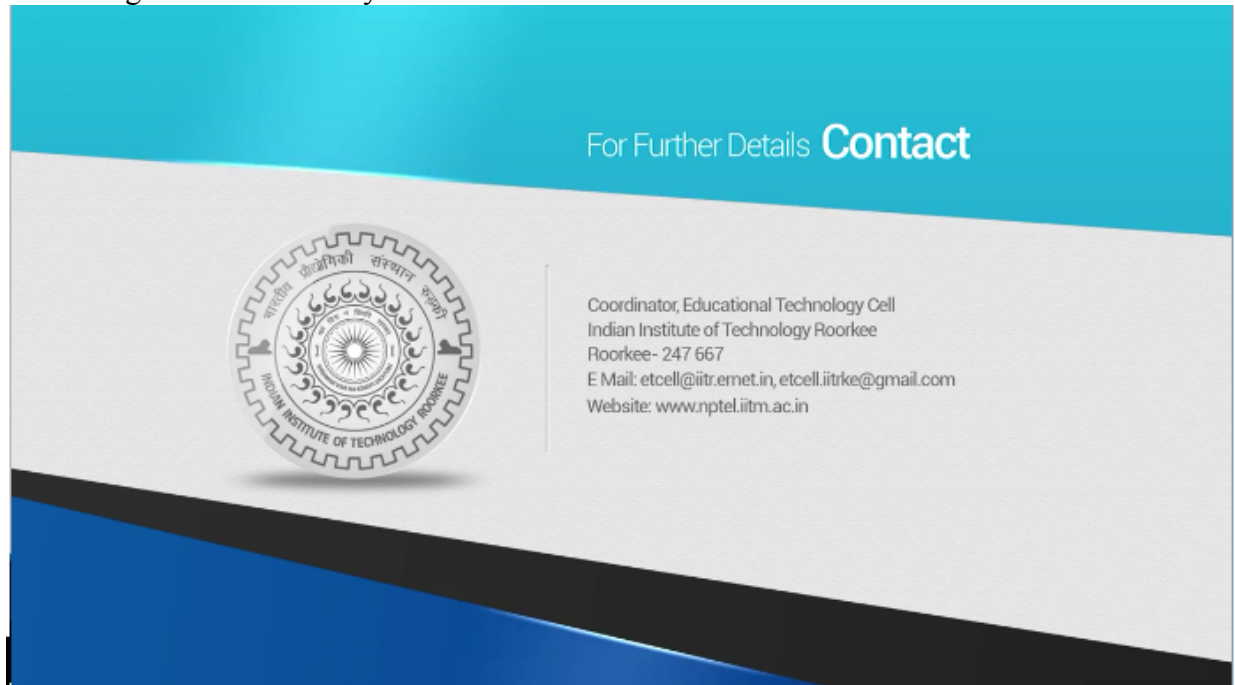


So what we are going to do is, we'll go back to R studio and do an exercise to find out whether a particular series is random walk or not, so for this we'll look at the S&P, BSE 500 monthly closing prices and we'll look at analyze whether this particular series is random walk, whether it makes sense for us to build a forecasting model for this series, so let's import this dataset.




So this is the structure of the series so we are going to create a time series vector here so you can see these are the values that we have, so we have 132 values from 2007 to 2018 and what we are going to do is here now we'll call this ARIMA function and we'll pass this time series vector that we have just created for this time series and you can see in the order we can clearly see we are just fitting the AR1 model, and once we do this we'll get the coefficient and this is the coefficient that we get, now you can see the value of this coefficient is 0.9931, value of the coefficient is 0.9931 this is very close to 1, so essentially if we go back to what we were

discussing this beta 1 that we've fitted the AR1 model and the beta 1 value is close to 1's, that means this BSE, S&P BSE 500 time series monthly closing prices, this seems to be a random walk therefore forecasting the series using any model might not be meaningful, right. So with this we'll like to stop here, and in the next lecture we'll start our discussion on smoothing methods. Thank you.



For Further Details **Contact**



Coordinator, Educational Technology Cell  
Indian Institute of Technology Roorkee  
Roorkee- 247 667  
E Mail: [etcell@iitr.ernet.in](mailto:etcell@iitr.ernet.in), [etcell.iitrke@gmail.com](mailto:etcell.iitrke@gmail.com)  
Website: [www.nptel.iitm.ac.in](http://www.nptel.iitm.ac.in)

For Further Details Contact  
Coordinator Educational Technology Cell  
Indian Institute of Technology Roorkee  
Roorkee – 247 667  
E Mail:-[etcell@iitr.ernet.in](mailto:etcell@iitr.ernet.in), [iitrke@gmail.com](mailto:iitrke@gmail.com)  
Website: [www.nptel.iitm.ac.in](http://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. Nibedita Bisoyi

Neetesh Kumar

Jitender Kumar  
Vivek Kumar  
Dharamveer Singh  
Gaurav Kumar

An educational Technology cell  
IIT Roorkee Production

© Copyright All Rights Reserved  
WANT TO SEE MORE LIKE THIS  
SUBSCRIBE