Time Series Modelling and Forecasting with Applications in R

Prof. Sudeep Bapat

Shailesh J. Mehta School of Management

Indian Institute of Technology Bombay

Week 02

Lecture 10: Practical Session in R-2

Thank you. Hello all. So, welcome to this new lecture in this course on time series modeling and forecasting using R. Again, in today's session, what we'll do is quickly revise a few properties about the time series process called random walk. So, from the last session, if you remember, we delved into multiple properties such as finding the variance function, finding the ACF function, or finding the auto covariances of some very basic time series models. For example, let's say autoregressive and then moving average, right?

And then we specifically took some lower orders of these two models, such as AR1 or MA1 and then MA2. And we established all these properties by working them out using pen and paper. And then, towards the end of the last session, if you vaguely remember, we talked about expanding a few more properties about the random walk. So, in today's session, we'll quickly revise those properties about the random walk, and then in the later half of the section, I'll try to bring in some ideas about how to plot a time series, how to visualize a time series using R, all right? So, initially, if you recall, this exactly is a random walk.

$$Y_t = \mu + Y_{t-1} + e_t$$

So, y t equals μ plus y t-1 plus et, and again, as always, et's are nothing but an IID sequence of random errors with, let's say, mean 0 and then some fixed variance σ square e, ok. And above all, this μ is nothing but the overall mean there. All right. So, again, if you notice this equation of a random walk, what I can do is keep on replacing these recursions, for example, let's say yt-1 or yt-2 and so on, in a recursive manner to expand it further, right. For example, what would happen if you write down yt-1? So, yt-1 would be nothing but let's say y t-1 equals μ plus y t-2 and so on and so forth, right. So, wherever you have a subscript in the past. So, let's say y t-1, that would be further expanded into y t-2 and so on and so forth. So, this would be equal to μ plus let's say y t-2 plus e of t-1, right. And then the eventual next step would be to expand this y t-2 further and further and so on.

$$Y_t = Y_0 + t\mu + \sum_{i=1}^{t} e_i$$

So, by recursion, one can actually get a neat formula for this time series process yt. And it could be kind of summarized in this manner. So, yt is what? So, yt is nothing but tμ plus the summation of all the error terms from 1 to t. So, tμ plus summation ej, with j going from 1 to t. So, this is an equivalent representation of a random walk. So, this is an equivalent representation of a random walk.

So now, what we'll do is we'll quickly look at some of the properties of this random walk. So, for example, the expected value or the mean of the random walk happens to be tμ, right? Now again, just for a second, if you go back a slide, this would be kind of evident from this alternative structure because you take expectations on both sides. Then what we will have, so if you take the expectation of yt, this would be nothing but tμ plus zero, right? Because the expectations of all the errors are zero.

$$E(Y_t) = t\mu$$

Okay, and similarly, if you take variances on both sides, then what will you have? So, on the left-hand side, you will have the variance of yt, and tμ is a constant, so the variance of any constant becomes 0, and then you will have the variance of summation ej. So, then you can bring out the summation and then take the variance inside, right? So, this would be something like the variance of summation ej, which would be nothing but the summation of variances of ej, and remember that each individual variance is equal to σ^2e, all right. So, eventually, this would be nothing but the summation of σ^2e, and this is probably what exactly you have on the next slide. So, the variance of yt happens to be γ0, which is nothing but t into σ^2e, because how many terms are there in the summation? You have exactly t terms.

$$V(Y_t) = y_0 = t\sigma_e^2$$

And now, the last important part is γk. So, how would you write down γk for a random walk? So, γk, if you remember from last lectures, is nothing but the covariance between yt and yt−k. So, here, I can replace yt. So, this guy is yt, and then this guy is nothing but yt−k. So, yt−k here, and then this is yt, and then the idea is to find out the covariance between yt and yt−k. Now, again, remember the technique I told you in the last couple of lectures that wherever you have common subscripts, those elements would add up to variances. For example, e1 and e1 or e2 and e2, right.

$$\gamma_k = Cov(Y_t, Y_{t-k}) = Cov(e_1 + ... + e_t, e_1 + ... + e_{t-k}) = (t - k)\sigma_e^2$$

So, for that matter, if you notice this carefully, all these subscripts from e1, e2, e3 up to e t-k occur in both places, right. So, eventually, what will you have? So, we will have t-k components of σ square e. Because exactly t-k elements or t-k subscripts are common between the first element in this covariance and the second sum in this covariance. So, eventually, the answer is nothing but t-k into σ square e, and all the remaining terms would be zeros because the subscripts are not matching.

Now, again, just one small disclaimer: the subscripts are not matching, so the covariance happens to be 0, and why exactly? Because all the errors are independent. So, for example, the covariance between e1 and e2 or e1 and e3, wherever the subscripts are not matching, such covariance happens to be 0. And now the last step is to find out rho k or the ACF function. So, by the way, this is nothing but the ACF function of the random walk. So, as per the formula, ACF is nothing but γ k divided by the square root of both the variances.

$$\rho_k = (t - k)\ \sigma_e^2 / \sqrt{(t\sigma_e^2\ (t - k)\sigma_e^2)} = \sqrt{1} - (k/t)$$

So, t-k into σ square e divided by the square root of t σ square e into t-k into σ square e, which sort of reduces to this function. And exactly the last line kind of gives you a few examples where one can actually try to model a random walk. So, for example, share prices, real exchange rates, or GDP of a country, and so on and so forth. So, all these are kind of valid examples where one can actually try to model these data sets using a random walk. And now the last thing that we have is a few simulations.

So here, you see a few simulations of a random walk. So, let's say on the left-hand side, we have a simulation where the overall mean μ is 0, and on the right-hand side, we have a situation where the overall mean is, let's say, 0.1. And immediately, you can sense that both these situations are completely non-stationary, right? So, in general, a random walk is supposed to be non-stationary, and why exactly non-stationary? Because, let's say, the mean depends on t or the variance depends on t. So, we saw these properties earlier, right? So, if you go back a slide, this is exactly what one can conclude from these two ideas.

For example, the mean depends on t, or the variance also depends on t. So, if something is dependent on time, which is t, then the process cannot be stationary. It has to be non-stationary. So, I guess this kind of sums up a few important properties of really basic time series models. For example, let's say a random walk or AR1, for that matter, or a moving average with orders, let's say one or two, right? And this sort of establishes a rather larger kind of situation where you analyze a few things or a few properties about all these basic ideas much stronger, okay?

Now, if you remember towards the end of the last lecture, I kind of gave you an idea as to how one can identify models based on the ACF plots and PACF plots, right? And then we kind of stopped the last session there. So, just to introduce this idea, and then we'll spend a few minutes here. This idea is kind of important to us also, that is, how do you identify some very basic models using the ACF plots and PACF plots, okay? Now, again, just to summarize this plot, or this window, or this slide. So, what you have is the ACF plots of two different models in the first row, and then you have the PACF plots of two different models, or the same models, on the bottom row, right.

Now, the left column gives you AR2, and then the right column gives you MA2, okay. So, an autoregressive model of order 2 and then a moving average model of order 2. Now, just by seeing all these four plots here, one can actually see some special things going on. For example, if you observe this ACF for MA2, which is given here. So, by the way, in each plot, if you notice, you have horizontal blue lines.

So, these horizontal blue lines represent some hypothetical confidence bands, right? So, if any spike is outside the bounds or outside these lines, we will say that that particular correlation, either ACF or PACF, is significant. So, possibly in this plot, the first correlation is significant, then the next one is significant, but after that, all the correlations are not significant because all the spikes are in between the blue lines. What do you mean by this is, let us say if you number the lags. So, the first spike is lag 1, the second spike is lag 2, then lag 3, and so on, okay.

So, what this plot is telling us. So, the ACF function of an MA2 process kind of cuts off after lag 2, and then this property is easily seen here, isn't it? So, at lag 1 you have a significant spike, at lag 2 you also have a significant spike, but after lag 2 all the other spikes are not significant. So, we'll say that the ACF function of an MA2 process cuts off after lag 2. And exactly a similar thing is observed in this PACF plot for AR2.

Can you see that? So, the same thing is happening here. So, let's say if you again number the lags, so this would be lag one, then lag two, then lag three, and so on. So, what this plot is telling us, the bottom left plot, is that the PACF function for an autoregressive model of a certain order, so let's say two here, cuts off after that order. So, one can actually get a cutting-off tendency from either the ACF plot or PACF plot of either an AR model of a certain order or an MA model of a certain order, right?

And you will see something special happening in these plots also. So, if you look at these plots, so the top left one and the bottom right one, what we can say is that these plots are, so you see a tailing-off tendency. So, tailing off means what? So, tailing off means that something is kind of gradually decreasing, right? So, tailing-off tendency in let's say this one, so this plot here and then the bottom right plot here.

So one can actually analyze all these four plots individually or for that matter one can analyze an ACF plot and the PACF plot so as to somehow get an idea about a possible model that might be fitted on the data. So, if you want to expand this in general, right? So, what one can say? So, one can say that if the ACF plot, right? So, if the ACF plot for an MAQ process, if you are analyzing an ACF plot for an MAQ process, then it has a cutting-off tendency.

So, the plot cuts off. So, the plot cuts off after which lag? So, after lag q, all right. So, if you want to identify an MA(q) process using the ACF or PACF plots, then one has to specify or focus on the ACF plot. And if he or she observes a cutting-off tendency in the ACF plot, then we can actually say that there is a high chance the model could be MA(q).

Similarly, if you want to identify a particular AR model of a certain order p, then one has to focus on the PACF plot, and an exactly similar structure can be seen here, right? So, again, just to remember or summarize this slide, identifying an AR model could be done easily using a PACF plot, while identifying an MA model could be done using an ACF plot, right? And just one last line to kind of leave you with this is that if you're trying to analyze any practical time series data and at an initial level you want to find out which model could be best suited for the practical data, then one has to plot both these and then check. For example, the ACF plot and the PACF plot, and the tendencies and the behaviors within those plots, to possibly suggest a good model for the underlying data, okay? All right, so now what we'll do is shift attention to a practical worksheet in R, and then we'll try to introduce some time series packages in R and work with some practical data to see how to plot it, right?

And then how to analyze some trends in the data and so on and so forth, all right? Okay, so here what we have is a worksheet in front of us which is called, let's say, code one, all right? And then if you look at the first line, so for solving this worksheet or for going through this worksheet, we'll be requiring one package in R, and this package is kind of inbuilt in R. So, the name is 'tseries'. Now, again, remember, probably we covered this in the first session also, but if not, I'll tell you once more that one can actually install any packages by going to the tools tab. So, if you see the tools tab and then install packages, and here one can actually type the name of the package.

For example, 'tseries', right, and then click on 'tseries' and then click on install. So, it will install the package, right? And then once you install the package successfully, you have to call the package in the R environment also. All right. So, that thing can be done using a library command.

So, once you install the package, you have to call the package in the R environment by using the library. Okay. So, this is exactly what we'll do. So, library t-series. And you have two ways of running any line in R code.

So, one can actually click on this run here, which is the easiest one. Or one can press control enter. So, control and enter. So, for this example, we'll be using a particular worksheet which gives you the US population data. So, if you see here on line nine, this is exactly how one can actually read the data in the R environment.

So, I'm giving the dataset a name, let's say US pop. And then what exactly is my USPOP? So, USPOP is nothing but read.table because my data is in a text file. So, since my data is in a text file, I will be reading the data from the text file. So, read.table and then USPOP.txt.

So, once you run this line, by the way, all that you are doing in the R coding window here would appear in the console window. So, the bottom left window is nothing but the console window. So, whatever outputs you create or whatever commands you process in the coding window would be kind of outputted in the console window. And then again, one good thing is to observe that if you don't see any errors here, it means that the code has run successfully. So, for example, as of now, we managed to load the t-series package in R and then also we managed to kind of read the US population data which was stored in a text file into R successfully.

Now again, you have two things. So, I think we discussed in the last lecture that one can actually specify header equals true or header equals false. Now, what do you mean by this? So, again, let me just show you the data. So, if you type in US pop here, then this is exactly the dataset that we have.

So basically, these numbers give you the US population on a yearly basis. So, initially, the population was that number. Then the next year, the population grew to that number. And finally, the last data point is exactly that. So, this is my dataset.

And then, how many elements are there? So, you have 21 elements. So, the length of the data or the sample size is nothing but 21. Now again, here, like I mentioned, you have two things. So, one can actually specify header equals false or header equals true.

Now, what do you mean by header? So, for example, if you look at this dataset, So the name of all the columns or the first row of any dataset is nothing but the header. So, the header gives you the names of all the columns that are in the dataset. So, if you specify header equals true, it will basically, so R would treat the first row as the column names and not the data itself.

I mean, so what might happen is that if you specify header equal to false, then this V1 that you see here might be treated as a data element by R. We don't want that, right? Because v1 is what? So, v1 is not a data point for us. So, v1 is a column name for us, right?

So, if you want to specify and if you want to tell R that R should read the first row in terms of column names, then it's always a good idea to specify header equal to true in that case, all right? So, for example here, right? If you again rerun this, so let's say uspop and then read.table, USpopulation.text while header is equal to true, then again it'll process the same data that we saw earlier, which is by the way given here, but now R will consider the first row as the column name or the column heading, okay? So, you have a small comment here that by default header equal to false is used, but what would happen if you use header equal to true, right?

And then one can actually use a scan function. So, the scan function is an alternative way to read any text file. So, let's say scan and then uspop.text. So, one can use either or. So, one can use read.table or the scan function.

So let me run this. So, again, immediately in the console you see that R has read 21 items. So, it's a good sign. Now, the next thing is how would you be able to plot the data. So, let us say if you want a very basic plot of the data set.

So, here we will use a plot function. So, the plot function is again an inbuilt function. So, one can use the plot function on any given data set. So, just for an easy idea or easy visualization. So, what we are doing here is that we are kind of dividing each value of the population by that number.

So, that my y-axis is not very elongated. So, I want to compress my y-axis, all right. And then these are some of the other technicalities that one should take care of inside the plot function. For example, type equals O. So, type equals O means it will kind of output O's instead of straight lines. And then main. So, main is if you want to give the plot a heading.

So, in our case, we are giving this heading to the plot. So, US population from 1970 to 1990. And then down the line, you have X lab and Y lab. So, X lab stands for the X-axis label. And then Y lab stands for the Y-axis label.

So, let's say the X-axis in our case is years. And then the Y-axis is nothing but the actual population in millions. And then the last command is LWD. So, LWD stands for the length or the width of the pointer. So, LWD stands for the width of the pointer.

So, how big or how small do you want all the pointers to be? So, LWD in our case is 2, basically. And here, you can kind of change the numbers here. So, let us say LWD could be 1, 3, 4, etc. So, now we will try to plot this.

So, if you try to plot this, basically, again, one can click on run here. By the way, on the bottom right corner, you see all the outputs, which are either plots or charts, and so on. So, this is a very raw plot. And one can actually zoom the plot. So, if you zoom the plot, this would be the actual plot. Now, let me extend this slightly, alright.

So, this is the plot that we kind of constructed now. If you see on the x-axis, we have all the years like this, and on the y-axis, you see the population values, and this is exactly how the plot looks like. Now, again, we have a small problem here that one can't find out what all these pointers mean as far as the x-axis is concerned. So, for that, what we can do is specify the x-axis. So, I can specify my axis to be something like this. So, I want a sequence from 1 to 21, starting from 1970.

So, the first point is 1970, or the first year is 1970, and the last year is 1990, okay. So, if you add this axis command to the plot now, if you see the plot again, it will give you the exact same plot, but now we have the axis numbering on the x-axis. Something like that. Now, let me try to zoom this slightly. So, now if you see that the x-axis contains all the years for us, and you have a heading for the plot, which says 'US population from 1970 to 1990' and so on. Now, just a couple of observations by looking at this plot is that the graph of the time series is not stationary.

So, this example is not stationary because you have a clear trend. So, let me close this graph and then move on. Now, if somebody wants, he or she can actually save the plot or save the graph as a PDF file. And one has this inbuilt command which is called PDF. And then you basically write down the name of the PDF file you want.

So, something like file equals USPOP.PDF. And then, if you run this, R would automatically save a PDF version of this plot that you got here in the same working directory. So, again, I can run the same commands. So, let's say plot and then access and so on and so forth. Now, the next idea is we will kind of try to play around with this generated plot.

So, let's say we'll try a transformation because, as we mentioned, this US population time series is not stationary. So, how do you make it stationary, basically? So, one idea is to apply some transformation. So, let us say a square root transformation. So, this root.uspop command is taking a simple square root transformation of all the population values.

Now, let us see what happens if you replot this. So, if you replot the same thing, then I will kind of get a slightly different plot along with the axis, of course. Now, let me zoom in. So, this plot is nothing but it gives you the square root origin of the same plot. So, if you notice the y-axis, things have not changed drastically.

The only thing that has changed is the y-axis has become slightly different now, right. So, the moment you take a square root, it kind of compresses the entire plot even further, all right. Okay, so now since you have discussed that the plot is not stationary, one can actually fit a linear line, right, because just by looking at this plot, if you remember vaguely the shape of the plot, I can actually fit a regression line or a straight line on all the points, because here you do not have a very severe curvature kind of thing, right. So, remember one thing, if the trend is looking more like a straight-line trend, then one can actually fit a straight line or a regression line on that to capture the trend. So, this is exactly what we will do.

So, for that, we require how many data points you have. So, R has an inbuilt command called length. So, the length of any data would give you the number of data points you have. So, here if you input, let us say, nt. So, nt would output 21 because we have 21 observations in the dataset, right.

And now the next line is kind of fitting a very simple linear model or a linear line on that example. So, again, we will make use of an inbuilt command called LM. So, LM stands for linear model. Now, here, what exactly is the response variable? So, the response variable is my population data, and then my independent variable is nothing but 1 to 21.

And then I am giving a name to this command, which is fit. So, if you run this, then it will basically fit a regression line on the population data. And from this, I can actually run a command called summary. So, the summary of fit would kind of give me a detailed output of the regression fit. So, something like this.

So let me... elongate this further. So, here you see what my residuals were, right? For example, how are the residuals distributed, or how exactly are the coefficients coming out? So, for example, what are my regression estimates, and what are the standard errors, right? And what are the corresponding p-values?

So here, if you see, the predictor is strongly trying to estimate the response because the p-value is significant, right? So, the p-value is so small that I can actually reject the null hypothesis and conclude that there is a relationship between my independent variable and the population values. And down the line, it gives you some other values also. For example, R-square. So, here, you see that the R-square value is very strong, almost 1, so 0.9921 and so on. So, these are some of the ideas that a summary command on the regression fit gives you.

Now what we can do is fit a straight line on the plot, so AB line, and then fit. So, if you run this AB line and then fit, now let me zoom in, okay? So, what this AB line applied on the fit would give you is, along with the plot, it will give you a straight line fitted on the plot. So, let me rerun this first. So, let's say NT and then my fit command, right?

And then if you want an AB line on that fit, it should actually output a straight line on that fit, basically okay. Now the next thing we'll do is try to remove this trend because we don't want the trend, right? So, how can we remove the trend? One can actually remove the trend by doing something like y minus y hat, right? Where y is the actual value and y hat is the fitted value, right. So, y minus y hat is expected to be somewhat stationary, right? Because once you subtract the fitted values from the actual values, one should expect that the resultant should be somewhat random process and stationary, right? So, this is exactly our approach here.

So, let us say, the actual population value minus the fitted values, and then if you run this, and then now if you want to replot this. So, how would the plot look like? So, my plot would look something like Something like that where you don't have a trend, basically. So, it will be more like a stationary kind of process.

Or, what I'll do is just for the sake of it, I'll delete this plot and then I'll try to replot a few things. So, by the way, this is the plot. So, let's say, this plot kind of gives you the y minus y hat values. So, once you subtract the fitted values from the actual values, my plot kind of looks like that. So, here, you see that the plot is not as non-stationary as it was earlier, right?

So now, you don't have an overall trend or something like that. So, this is the idea. So, one can actually try to fit a regression line and then find the fitted values. And then, if you try to plot y minus y hat kind of structure, then it should kind of resemble a stationary process, right? And once you arrive at this, then one can actually find out multiple things of that resultant, right?

For example, mean. So, in our case, the mean is close to zero, if you see, or the variance. So, the variance is 0.15 or the standard deviation, and so on and so forth, right? Now, if you want, I can actually add a straight line at the mean to my plot also. So, probably something like that.

So, just to tell you exactly where the mean lies, basically, right. So, once you plot y minus y hat, my mean is so close to 0 that you have a horizontal red line at 0, right. So, I guess these are a few things that one can do, I mean, start doing once you get hold of some practical data. So, the first thing is to plot the data, then identify whether the data has some non-stationary structure or not. So, maybe apply some transformation.

And then maybe try to plot the complete data along with the x-axis labels, y-axis labels, and a suitable title for the plot. And so on. And then, depending on how strong the trend is, right, one can actually try to fit a regression line. So, the trend is not that strong. So, if you have a straight-line trend, then why not just try to fit a straight-line regression and then kind of get hold of the fitted values, which are y hat.

And then, eventually or subsequently, subtract y minus y hat and then get hold of a stationary process. Okay. All right. Thank you. Thank you.