# Laboratory #4 Questions and Discussion Part 1 – Foundations

## The Overall Time Series Analysis and Forecasting Process

The first part of this is a bit long. I’ll put everything together is a slightly different way. This is intended to get you to think about the same things from a different perspective. Sometimes that helps in developing you intuition in what is going on.

First, let’s look at the big picture, i.e. what you are going to do. Your textbook has this in a couple places in different ways. Graphically, the textbook has the figure below in Section 5.1.

Specify

Visualize

Estimate

Tidy

Forecast

Evaluate

However, in Section 1.6 the textbook authors list the “basic steps in a forecasting task” (Hyndman & Athanasopoulos, 2021) as:

1. Problem Definition
2. Gathering Information
3. Preliminary (exploratory) analysis
4. Choosing and fitting models
5. Using and evaluating a forecasting model

Before this all starts you need to come up with your research question or problem statement. This is an essential part of any research project and is Step 1: Problem Definition. That sets up everything else you will do and is before you start the time series analysis and forecasting process shown in this graphic. Once you have that you’ll need to acquire your data, either acquiring your own data or someone else’s. Once you have data then you’ll need to pre-process it. That involves putting it in “tidy” format and visualizing the data to make sure that you can use it to answer your research question or address your problem statement. This is Step 2: Gather Information. “Specify” and “Estimate” are really outputs from Step 3: Preliminary (exploratory) analysis. Once you have preprocessed your data (Step 2) and conducted your exploratory data analysis (EDA), then you are ready to start your confirmatory data analysis or CDA. This is Step 4: Choosing and fitting models or “Evaluate” on the graphical process shown. During your evaluation you will fit models, run diagnostics and determine the accuracy of your model(s) or how well the model(s) fit the data. If you find that your model works well then you’ll want to break out of the time series analysis process circle and go to Step 5: Using and evaluating a forecasting model. That is you will make a “Forecast” using the model or models you have determined work well for your research project. I know that Step 5 also includes the word “evaluating” but you will want to evaluate your forecast to make sure it is sufficiently accurate for your project too. So, it all fits. You may want to refer back to this time series analysis circle and/or the steps involved in conducting a time series analysis and in forecasting from time to time, just to keep yourself on track. With all the details in each step or “phase” of the process it can be easy to get lost in all the trees, losing sight of the overall forest! Meanwhile getting back to our labs, I also want to take some time to go over the things we’ve been covering in a couple different ways. My purpose is to help you develop your intuitive understanding of what is going on when you conduct a time series analysis and do your forecasting.

Labs 1 and 2 were all about skills-building. By the end of Lab #2 we began to see that there were three basic types of time series data; stationary time series data, non-stationary time series data, and cyclical time series data. Remember that cyclical data are data that have peaks and troughs but those peaks and troughs do not occur at periodic intervals. Lab #3 began developing your ability to conduct time series analysis and forecasting. However, Lab #3 began with time series analysis which was somewhat limited because it required stationary data. When we get to ARIMA(p, d, q) models we no longer will need our data to be stationary! But let’s take a few minutes and review exactly what we mean by stationarity or stationary data and autocorrelation (Dotis-Georgiou, Autocorrelation in Time Series Data, Retrieved 22 April 2025 from <https://www.influxdata.com/blog/autocorrelation-in-time-series-data/>).

## Stationarity

Stationarity is a property meaning that the “something” being considered has specific characteristics or attributes that cause it to be stationary. In the case of stationarity the characteristics are that the statistical mean, variance and autocorrelation (Singh, 2024). And, for stationarity to exist all these characteristics must be constant. That is, each is constant not that all are constant with each other. For example, a constant mean describes an unchanging level of a time series, or a constant variance says that a time series’ variability does not change. Take a minute and think about these statistical characteristics and what they really mean in time series analysis. If a constant mean is required then we cannot have a trend. Many, many things have trends. For example, if you are involved in sales you want an increasing trend which indicates increasing sales or increasing profits and so on. If a constant variance is required that can be a bit difficult to see from a time plot. I’ll talk about that a little more below. Autocorrelation is kind-of a strange thing when you really think about it.

One of our fundamental assumptions in conducting regression is independence. But, autocorrelation is a measure of the value of something and its similarity to a previous value of itself, i.e. its serial fdependence, which means it isn’t independent – oops! We use the same mathematics we previously used to determine the correlation between two different variables. But in this case we use a lagged version of a variable with itself (over successive time intervals) and compute its correlation. That is where the notion of “serial dependence” comes from. Autocorrelation has also been called serial correlation, time series correlation or lagged correlation. This is why the first regression models we developed for time series data relied on stationarity. We will soon start using code that accommodates non-stationarity in different ways. But you will still want to have these methods, achieving stationarity through differencing or transformations. Some put detrending in a different bucket than differencing. I’m just going to say detrending is part of differencing. The common tests for stationarity are the Augmented Dickey-Fuller (ADF) test or the Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test. These are often referred to as “unit root tests”. We will look at some additional diagnostics that can indicate that we have achieved stationarity but these are the actual tests that confirm that. Or, said another way, these are the tests that tell us whether or not we need to use some method or technique to make our data stationary. Unfortunately, such tests may produce conflicting results. In that case you need to really understand what is going on in the data in order to determine which of the conflicting results is meaningful.

To look at this from a more advanced development and give you even more information about stationarity, I’ve downloaded some lecture notes by Dr. Rauli Susmel an Associate Professor of Finance from the University of Houston. This will give you yet another way to think about stationarity and the statistical background for it. These are lecture notes from a second course in the PhD in Finance curriculum, PhD Econometrics II Course. But, they are directly relevant to what we are covering too. I’m pasting below two slides from Lecture 13 that illustrate how a more advanced course thinks about the topic of “dependence” which we have been approaching as a topic about “independence”. The website for these lectures is <https://www.bauer.uh.edu/rsusmel/> then follow the links.

The first slide (below) mentions that in “classical statistics” we look for “iid” or independent and identically distributed realizations of the value of the dependent variable at time . This is the direction we come from in our courses whether it is ANA 500, 510 or 535.



However, in time series where we expect future values of to be at least somewhat dependent on current values, we have a problem with “independence” because that means our measurements are “dependent”. So, we start talking about stationarity which brings us to the next slide in Susmel’s lecture shown below. Because this is a more advanced course prerequisites for it are more advanced, i.e. a more advanced level of statistics is required. Rather than the focus being on least squares regression, the focus is on maximum likelihood estimation (MLE). Regardless of the statistical process the end results will be the same.



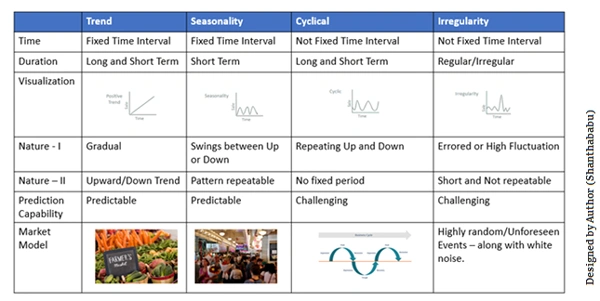
I believe that “DGP” in this slide refers to distributed Gaussian processes. Ergodicity is an interesting principle. We barely touched it in ANA 510. In this context, i.e. LLM (law of large numbers) and CLT (central limit theorem), it simply means that the average or mean outcome over time for an individual, e.g. a coin flip, will be the same as the average or mean outcome across a group at one single point in time. When he talks about the moments of the first moment is the mean, the second moment is the variance, the third moment is the skew, the fourth moment is kurtosis, and so on. It says the same thing we’ve been saying, constant mean and constant variance. And so on.

## Periodicity

Before we start considering how to use more advanced methods to build time series models there is one more connection to make with regard to the concepts we have already covered. That is the relationship between autocorrelation and periodicity (Lab #2). Many have written that “the usual method for deciding if a signal is periodic and then estimating its period is the autocorrelation function” (Eurospeech 2001 – Scandinavia, Retrieved from the Autocorrelation webpage at [https://spandh.dcs.shef.ac.uk/ed\_arena/mad/mad/auto/auto.htm 22 April 2025](https://spandh.dcs.shef.ac.uk/ed_arena/mad/mad/auto/auto.htm%2022%20April%202025)). But there are challenges with the autocorrelation function in that it can retain too much information making it difficult to pick out primary peaks. Don’t worry too much about this. You should just be aware that the real world is really a very messy place.

Season and Trend

One of the first techniques we specifically considered for time series analysis was STL(), or seasonal decomposition of time series by Loess. Using STL we decomposed time series data into its trend, seasonal and residual components where residual is the random fluctuations that are “leftover” or cannot be attributed to trend or seasonal patterns. STL is a non-parametric method meaning that the method learns the relationship between variables from the data. (On the other hand, a parametric method assumes a specific, known relationship between dependent and independent variables, etc., e.g. linear regression.) You begin using STL by decomposing the time series data and then removing the seasonal component. The decomposed time series can then be further used in forecasting. The issue you are addressing using STL and differencing via decomposition is non-stationarity. One of the tools you have to determine if you have been successful in achieving stationarity is the gg\_tsresiduals() command which shows the residuals plot, a histogram of the residuals and the ACF. With these diagnostics you can evaluate if you have been successful in achieving stationarity. The figure below summarizes the characteristics in time series data that we may have to deal with (Pandian, 2025).



There are several ways of addressing non-stationarity depending on what is causing the non-stationarity (Sanderson II, 2023). For example, a trend in data will result in a mean that is not constant. Differencing can be used to detrend time series data. If the data have a periodicity such as seasonality, differencing again but this time deseasonalizing rather than detrending the data may produce stationary data. Sometimes, if you have data with both seasonality and trend, deseasonalizing the data first may also remove enough of the trend to consider the data stationarity. Another way of handling non-stationary time series data is by using transformations, e.g. power transformations, square root transformations, or log transformations, where the log transform is the most common. The transformation you choose will have to do with the cause of the non-stationarity.

You have used the log transformation in prior courses but to linearize a variable such as price or wages. For time series data your textbook has several different cases where log transformations are applied. I’ll let you search on “transformation” to find those in your textbook. I’ve looked at these and believe several are very good. One I am concerned with is the Google stock price example(s). This is a complicated time series. It has several discontinuities similar to our Amtrak data. I believe that a log transformation might apply but you would need to approach the overall problem as piecewise continuous, e.g. as taking before 2019 and after 2022 in our Amtrak data. Then we did this and compared the forecasts we saw that the forecast changed if you used the data before the pandemic or after the pandemic. So, using a piecewise continuous approach you would need to look at what the forecasts were before and after a discontinuity and carefully merge those into the most accurate forecast. Once your data is stationary you can use a variety of model functions. Or, if desired you can apply more advanced model functions, e.g. the ARIMA function, without making your data stationary at all. I won’t include all the different smoothing methods right now, moving averages or exponential smoothing. I’ll just move onto the first models we considered.

## ets() Models

One of the first models we started building was the ets() model or error, trend and seasonality model. One of the benefits of an ets() model is that any of several well-known criteria can be used for model selection, e.g. AIC, AICc and BIC. These are discussed in Section 7.5 of our textbook. Akaike’s Information Criterion or AIC, the corrected AIC, or the Bayesian Information Criterion or BIC are all also used in machine learning. So, it is worthwhile to understand what they represent. These are alternatives to using a more simple method such as R-squared. For example, the idea behind AIC is similar to the adjusted R-squared which adjusts R-squared for the number of parameters or variables. AIC penalizes the fit of the model (SSE) with the number of parameters or variables. You can find out more about each of these on your own.

ets() models are statistical models with underlying exponential smoothing methods. In your textbook, Chapter 8 and in particular Section 8.4 begins to put together all the various exponential smoothing methods that can be used. For example, simple exponential smoothing is represented by (N,N). Holt-Winters’ method can be additive, multiplicative or damped. In Section 8.4 you will find all of this put together for you, from simple exponential smoothing models to double or even triple exponential smoothing models such as the Holt-Winter’s method.

## Time Series Models

Phew! That’s already a lot to think about and we are just now getting to the topics covered in Laboratory #4 ARIMA Models (Peixeiro & Updated by Urwin, 2023) (Wainaina, 2023). First, I’m sure you’ve already got this but let’s cover some terminology. In the acronym ARIMA, AR stands for Autoregressive. MA standards for moving average. And, the “I” in the middle stands for Integrated. In addition to ARIMA models, there are also SARIMA models as well as ARCH (Kenton, 2024) and GARCH (The Investopedia Team, 2024) models, and others from machine learning. A SARIMA model is a Seasonal Autoregressive Integrated Moving Average model. You can probably already guess some of what is involved in a SARIMA model. We won’t cover those or the ARCH or GARCH models. To be thorough, an ARCH model is a Autoregressive Conditional Heteroskedasticity model whereas a GARCH model is a Generalized Autoregressive Conditional Heteroskedasticity model. An ARCH model considers how the variance of the time series data changes over time. It focuses on periods of increasing or decreasing volatility. A GARCH model is very similar to an ARCH model but is used to focus on the volatility of financial returns on assets. For example, you might you a GARCH model if you are looking at the risk and expected returns on assets during periods of volatility. One last general note about selecting the “best” model, i.e. how do you know when you have the “best” model. “A time series model is adequate (good enough) if residuals are white noise”.

## Autoregressive AR(p) Models

Moving back to ARIMA models, we will look at AR(p) models, MA(q) models, non-seasonal ARIMA models or ARIMA(p, d, q) models where the argument “d” is the degree of first differencing used. Before we get into the more general models, let’s first look at models that represent common patterns we see in time series data. Let’s look at AR(p) models (Hyndman & Athanasopoulos, 2018) (Penn State, Eberly College of Science, 2024). For example, an AR(0) model is used to represent white noise. White noise refers to time series data in which there is no inherent pattern, i.e. it is “iid” with no trend, no periodicity or cyclical nature to the data. White noise typically has a mean of zero (0) and a constant variance . An example of time series data representing white noise is shown in the figure below.

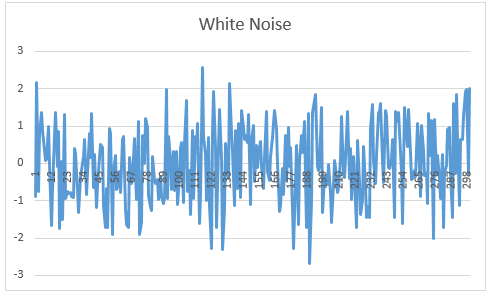


Figure 1 – White Noise Simulation (Zaiontz, 2022)

Higher order AR models such as AR(1), AR(2), etc. can use this model of white noise (AR(0)) as the error term. This “white noise term” allows a model to capture the essence of real-world stochastic processes. Essentially, white noise is a basic building block of ARMA models.

Let’s take one step back and pick up other parameters in AR(p) models. The equation of an AR(p) model is given by

where is white noise. Let’s consider the coefficients . Changing also changes the pattern of the time series data. Changing only changes the scale of the series. Let’s look at a couple different values of . From Section 9.3 in your textbook, is white noise (Hyndman & Athanasopoulos, 2021). When you have a random walk. A random walk really is what its name implies. A random walk is a stochastic (random) process with each future value determined by the current value plus a white noise term. If you start at and add the white noise terms as shown in the figure above you would get the same figure back. If you ran a white noise generator again you would get something different. It is completely random. When then you have a random walk with drift. The drift accommodates any trend, increasing or decreasing. When then oscillates around the mean.

Because AR(p) models are usually restricted to stationary data, remember the “I” in ARIMA is what allows you to use non-stationary time series data, then we have to restrict the parameters. For example, when we want an AR(1) model (remember that the current value of is strongly correlated with the prior period value of , then . Or said another way, the argument “p” for an AR(p) model is related to the lag in the model. For example, if time series data are strongly correlated with prior period data the lag equals 1 (lag = 1). Then, the argument “p” will also equal one (1) or AR(1). If we want an AR(2) model then . When , things get a lot more complicated. Some applications for AR(p) models are signal processing and natural language processing (Jindal, 2024).

## Moving Average MA(q) Models

Let’s look at MA(q) models for a few minutes. We considered moving averages as a smoothing technique before (Penn State, Eberly College of Science, 2024). It is unfortunate that the same words are used for this type of model because these models don’t have much to do with the moving averages we applied before. Whereas an AR(p) model uses past values such as , a MA(q) model uses past values of forecast errors. And, where the parameter “p” in the AR(p) model is related to the number of lags to include, the parameter “q” in the MA(q) model is the number of lagged error terms to include. It’s a bit more complicated in the way forecast errors are applied in MA(q) models. In an MA(q) model each value of can be thought of as a “weighted moving average” of the past forecast errors (Hyndman & Athanasopoulos, 2021). The general equation of the model, as shown in Section 9.4 of your textbook is (Hyndman & Athanasopoulos, 2021):

Again, we use white noise for which only changes the scale not the pattern. One interesting situation for MA(q) models has to do the ACF. Consider the equation for the ACF:

* Mean is
* Variance is
* Autocorrelation function (ACF) is:

That is, the only nonzero value for the ACF is for lag = 1! So, if you plot the ACF and the only significant ACF spike is a lag = 1 you probably have or can use an MA(1) model. What about an MA(2) model?

* Mean is
* Variance is
* Autocorrelation function (ACF) is:

Similar to the MA(1) case, for MA(2) the only nonzero, significant ACF values are for lags 1 and 2. So, if you plot the ACF and the only significant ACF spikes are at lags 1 and 2 you probably have or can use an MA(2) model. As was the case for AR(p) models with things get much more complicated (Penn State, Eberly College of Science, 2024) (Hyndman & Athanasopoulos, 2021). (Note that this citation isn’t quite correct. It should be (Penn State…, 2024; Hyndman…, 2021) all in one set of parenthesis separated by a semi-colon. But Word doesn’t allow me to do that. I wouldn’t take any points off for this sort of thing as long as you have cited all the sources. But you should know I’ve seen other faculty do so.)

Another interesting property of MA(q) models is that they can be invertible. If you’ve had linear algebra you know that this is a good property to have. If you have an MA(q) model that converges to an AR() model it is invertible. In this case, all the AR coefficients tend to zero (0) as we move back in time, i.e. t-p where p is large and gets increasingly larger.

One question you might be asking is, “Why use an MA(q) model?” Well, there are several advantages, given the situation you are trying to model, to use an MA(q) model. For example, because MA(q) models use lagged error (or noise) terms and we know that noise terms are typically have a distribution with a mean of zero (0). By using past forecast errors, MA(q) models can produce more accurate predictions and can be more easily interpreted, particularly if short-term fluctuations are present. This makes MA(q) models great for financial forecasting. Another use of MA(q) models is anomaly detection. But keep in mind that the short-term fluctuations should not highly volatile (or represent enormous changes) like we saw in our Amtrak data during the pandemic. Further, because MA(q) are great in establishing baseline behavior, because of their reliance on past forecast errors that have distributions about zero (0), any outliers or unusual patterns that deviate from the baseline will be obvious (Borah, 2024).

## ARIMA(p,d,q) Models

Finally we can talk about ARIMA(p,d,q) models. You already know that AR stands for Autoregressive and MA stands for Moving Average. The “I” in ARIMA stands for integrated. Essentially that is the number of differences taken to achieve stationarity hence the parameter used is labeled as “d”. When differencing is applied to detrend time series data we can build a “non-seasonal” ARIMA model. This is covered in Section 9.5 of your textbook (Hyndman & Athanasopoulos, 2021). Our textbook talks a bit about maximum likelihood estimation (MLE) in Section 9.6 or the above information from the Susmel at the University of Houston. One big question is whether or not you can run a code that will automatically optimize an ARIMA(p,d,q) model? The answer is yes! The fable package in R allows you to do that. It uses the corrected AIC to determine what the best optimized model is. Section 9.7 in our textbook discusses how this criteria is applied to determine the “best” model. On the other hand, Section 9.9 in our textbook describes how to evaluate the ACF and PACF in order to setup the parameters p, d, and q by hand. If you do allow the code to automatically setup the parameters, you can print out what the code finds for the criteria. Lastly, Section 9.10 in our textbook compares ets() and ARIMA() models. For the last plot we generate in Laboratory #4 we will compare lm(), ets(), and ARIMA() models.

Section 9.7 of our textbook includes a graphical representation of the overall time series analysis and forecasting process. To complete the steps in Lab #4 you will both select the value of the parameters of models yourself and you will let the code automatically optimize the values of the parameters based on the corrected AIC. Now, let’s get to the actual questions to answer for Laboratory #4!

## Step 1 – the Fundamentals

Step 1 comprises several sub-steps, each intended to walk you through the fundamentals behind ARIMA(p,d,q) models that we discussed above.

### Step 1a – AR(0) Models

Time series data that an AR(0) model fits well are generally referred to as creating a pattern known as white noise. Those data have no predictability or periodicity. They are completely random. Use a mean of zero (, a variance of 4.0 (or a standard deviation of 2.0) to generate data. We can generate these random time series data with a specified mean and variance using the rnorm() function. I’ve included that in the Lab4Step1 script.

Follow the process outlined in Section 9.7 to conduct your time series analysis on the data you have generated. Note that I would advise against using a Box-Cox transformation unless you really need to. If you need to, do it. However, remember that any transformation you do you will also have to “undo” to get your final answer in real-world units such as dollars and cents.

For Step 1a we computed the ACF and plotted it. Because we knew the data we generated would be white noise we anticipated there would not be any significant spikes in the ACF, and there was not.

1. Based on the time plot of data generated for Step 1a, can you tell if there is any trend or periodicity in the data?
   1. Yes
   2. No x
2. A time plot is made from the following type of data object, \_\_\_\_\_\_\_\_\_\_ .
   1. time series data x
   2. panel data
   3. logical data
   4. floating point numeric
3. The Autocorrelation Function for white noise will have \_\_\_\_\_\_\_\_\_\_ .
   1. no significant spikes x
   2. a significant spike at lag = 1
   3. a significant spike at lags = 1 and 2
   4. none of these choices
4. Unit root tests include \_\_\_\_\_\_\_\_\_\_ .
   1. the Augmented Dickey-Fuller test
   2. the Kwiatkowski-Phillips-Schmidt-Shin test
   3. the ADF test
   4. the KPSS test
   5. all these choices x
5. If the p-value for the Augmented Dickey-Fuller test is less than 0.05 and the p-value for the Kwiatkowski-Phillips-Schmidt-Shin test test is greater than 0.05 this means \_\_\_\_\_\_\_\_\_\_ .
   1. the results of both tests are consistent x
   2. the results of the ADF test mean the data are stationary
   3. the results of the KPSS test mean that the data are non-stationary
   4. none of the above choices

### Step 1b – AR(1) Models

1. You would first plot the data in order to determine if there were any \_\_\_\_\_\_\_\_\_\_ .
   1. outliers in the data
   2. patterns in the data (such as periodicity or a trend)
   3. unusual observations (such as the precipitous drop in the Amtrak data due to the pandemic)
   4. all these choices x
2. The second step in the time series analysis process is usually \_\_\_\_\_\_\_\_\_\_ .
   1. making a transformation on the data
   2. generating a tsibble from the data
   3. determining if the data are stationary x
   4. none of these choices
3. To determine if the data are stationary you could \_\_\_\_\_\_\_\_\_\_ .
   1. conduct a unit root test
   2. decompose the data x
   3. fit a model to the data
   4. none of these choices
4. The different ways to make non-stationary data stationary include \_\_\_\_\_\_\_\_\_\_ . Wrong
   1. differencing the data xx
   2. using a transformation such as taking the log of the data
   3. detrending and/or deseasonalizing the data
   4. all of these choices x
5. A plot of the Autocorrelation Function for data that an AR(1) model fits well has \_\_\_\_\_\_\_\_\_\_.
   1. no significant spikes anywhere
   2. quickly (or exponentially) decaying spikes beginning at lag = 1 x
   3. one significant spike at lag = 1
   4. all of these choices
6. A plot of the Partial Autocorrelation Function with one significant spike at lag = 1 means \_\_\_\_\_\_\_\_\_\_. Wrong
   1. that there is no correlation anywhere in the data, the spike is a mathematical artifact
   2. that there is a correlation in the data but without further testing you cannot tell where
   3. that there is a correlation in the data at lag = 1 (the prior period) x
   4. none of these choices xx
7. To use an AR(1) model the data must be \_\_\_\_\_\_\_\_\_\_ .
   1. a large data set (at least 1000 observations)
   2. periodic in nature
   3. stationary x
   4. all of these choices
8. Unit root tests for the data that an AR(1) model fits well indicate that the data are \_\_\_\_\_\_\_\_\_\_.
   1. non-stationary
   2. stationary x
   3. periodic
   4. linearly, increasingly trending

### Step 1c –

### Step 1d – MA(1) models

1. To use an MA(1) model the data must be \_\_\_\_\_\_\_\_\_\_ .
   1. a large data set (at least 1000 observations)
   2. periodic in nature
   3. stationary x
   4. all of these choices
2. Unit root tests for the data that an MA(1) model fits well indicate that the data are \_\_\_\_\_\_\_\_\_\_.
   1. non-stationary
   2. stationary x
   3. periodic
   4. linearly, increasingly trending
3. The Autocorrelation Function for data that an MA(1) model fits well has \_\_\_\_\_\_\_\_\_\_.
   1. no significant spikes anywhere
   2. one significant spike at lag = 1 x
   3. two significant spikes at lags 1 and 2
   4. none of these choices
4. A plot of the Partial Autocorrelation Function with one significant, positive spike at lag = 1 and a negative, significant spike at lag = 2 means \_\_\_\_\_\_\_\_\_\_. Wrong
   1. that these data likely represent white noise
   2. that these data are non-stationary
   3. that there is a correlation in the data at lag = 1 (the prior period) xx
   4. none of these choices x

### Step 1d – AR(1) with drift

1. Based on the results of the ADF test and the KPSS test, the data that an AR(1) model with drift fits well are more complicated to analyze. That is because the \_\_\_\_\_\_\_\_\_\_ and the \_\_\_\_\_\_\_\_\_\_. Wrong
   1. ADF test indicates the data are non-stationary; KPSS test indicates stationarity xx
   2. ADF test (and the) KPSS test both indicate the data are non-stationary
   3. ADF test (and the) KPSS test both indicate that the data are stationary
   4. none of these choices x
2. The data for Step 1d that an AR(1) model with drift fits well are likely non-stationary because \_\_\_\_\_\_\_\_\_\_. Wrong
   1. the drift creates periodicity in the data
   2. the drift creates a cyclical pattern in the data
   3. the drift creates a trend in the data x
   4. none of these choices xx

### Step 1e – Earthquakes Data

1. The biggest problem with the quakes data is that the time intervals between observations are not equal so any time series analysis conducted without taking care of this by interpolation, resampling, state-space representation or some other means of making the time intervals equal.
   1. True x
   2. False
2. The ACF for the quakes data looks a lot like an \_\_\_\_\_\_\_\_ model would fit it well.
   1. AR(0)
   2. AR(1) x
   3. MA(1)
   4. none of these choices

### Step 1f – Google Stock Closing Price data

Use the Google Stock data to complete this step. These data were retrieved online from <https://www.nasdaq.com/market-activity/stocks/goog/historical?page=1&rows_per_page=10&timeline=y5>. They include the Google Stock data for the past 5 years.

1. Based on a time plot of the Google data, those data have a \_\_\_\_\_\_\_\_\_\_ .
   1. monotonically increasing trend
   2. monotonically decreasing trend
   3. a flat trend
   4. none of these choices x
2. The ADF test result that the p-value equals 0.6336 (and hence we reject the null hypothesis) indicates that \_\_\_\_\_\_\_\_\_\_ .
   1. we accept the null hypothesis that the data are non-stationary x
   2. we reject the null hypothesis and accept sufficient evidence that the data are stationary
   3. based on the time plot we believe the data are non-stationary even though the ADF test results indicate that the data are stationary
   4. none of these choices
3. The ACF plot is \_\_\_\_\_\_\_\_\_\_ .
   1. very strange looking and needs to be redone
   2. consistent with the results of the ADF test, the data are non-stationary x
   3. a higher order AR(p) model will be required
   4. none of these choices
4. The PACF plot indicates \_\_\_\_\_\_\_\_\_\_ .
   1. that there are inconsistencies and work may need to be done to make the data stationary
   2. some inconsistencies and that an AR(1) model may fit the data well x
   3. that there are correlations in the data at lags 1 and 12
   4. none of these choices

### Step 1g – Check the automatic ARIMA fit

1. After running the code to automatically fit an ARIMA model what are the parameter values you get for the e2 data?
   1. 0,0,0
   2. 1,0,0 x
   3. 0,1,0
   4. 0,0,1
2. This is consistent with the values you found for the simulated data e2 before.
   1. Yes x
   2. No
3. Consider the e3 data intended for an MA(1) best fit model. Remember that we generated simulated data that an MA(1) model would fit well. Now, generate models for an ARIMA(0,0,1) model as well as the automatically generated ARIMA models. Although the values for AICc are very close, which has the most accuracy in terms of AICc?
   1. ARIMA(0,0,1) x
   2. ARIMA(0,0,0)
   3. ARIMA(0,0,3)
   4. ARIMA(3,0,0)
4. This is consistent with the values you found for the simulated data e3 before. Wrong
   1. Yes x
   2. No xx
5. Now look at the earthquake data again. Recall that considering the ACF and PACF that we thought an AR(1) model would fit the data well. Run an ARIMA(1,0,0) and let the code automatically generate ARIMA models. Again considering the AICc values, which model is the most accurate? Wrong
   1. ARIMA(1,0,0)
   2. ARIMA(0,0,0)
   3. ARIMA(0,0,3) xx
   4. ARIMA(1,0,1) x
6. Although the quakes data was more difficult to interpret due to the uneven time intervals, this result is consistent with the best model that was found earlier for the quakes data.
   1. Yes
   2. No x
7. Lastly, consider the Google stock closing price data again. Like was the situation with the quakes data, the Google data has several missing values. However, based on the ACF and PACF we found what would or could be the best model, i.e. an AR(1) model but it looked like there should be a difference taken. Again using the code available in fable, run the ARIMA(1,1,0) model, since the fable code produces an ACF and PACF that has a significant spike at lag = 3 run an ARIMA(3,1,0) model, then run the automatic ARIMA models. Which of the following ARIMA models is **least** accurate using the AICc criterion?
   1. ARIMA(1,1,0) x
   2. ARIMA(3,1,0)
   3. ARIMA(2,0,3)
   4. ARIMA(2,0,4)

(Note that I had to make this question ask about the least accurate answer because the other three were actually in a 3-way tie for most accurate.)

In fact for all the models for all these different datasets, there isn’t that much difference in the overall levels of accuracy as measured by the AICc criterion.

This is the end of the questions for Step 1, Foundations.