

Time Series Econometrics

Justin Raymond S. Eloriaga
De La Salle University
School of Economics

First Edition
2020

For my Dad up there
May the Force be with you, always

Preface



*"I find your lack of faith disturbing" - Darth Vader
Star Wars Episode IV: A New Hope*

The very first movie my dad and I watched together on old VHS was Star Wars Episode IV: A New Hope. He said to me that it was going to be the best movie ever. And he was right. I always believed that the original Star Wars was the best of all. Call it nostalgia or the inner kid in me, but it was something truly special. As any kid would do, I imitated the big bad villain, Darth Vader, on numerous occasions to try and get some humor out in order to impress my dad. You see, I wanted a lightsaber of my own, hence, the flattery my six year old self would reach to achieve my goal would leave you truly breathless. Note that the price of the cheapest lightsaber that time was around ₱2,500. Adjusted for inflation, that's roughly ₱3,500 to ₱4,000 or about \$100. I, of course, succeeded in getting one, even buying the exact lightsaber of Lord Vader himself. In the movie, one line truly stuck with me which was Vader's response to one of the grand admirals who had questioned him. Vader uttered the now memed-line of *"I find your lack of faith disturbing"* in his infamous low ghostly voice. I remember that whenever I found something too difficult or impossible and rant about it, my dad would always say those lines to me, very much intentionally. This would then trigger some fond childhood memories. Let's cut to the chase. Time series econometrics is not an easy subject or course. Heck, many people purposely avoid it at all costs. What I offer you is this, heed the advice of Darth Vader. You wouldn't want to be force choked to death. By all accounts, have faith so you wouldn't be so disturbed. May the force be with you, always.

This is a compilation of my notes in **Time Series Econometrics**. I have constructed it in such a way that it approaches the subject matter rigorously but in a more palatable and approachable manner. As this is just my personal work, there are bound to be errors to which I apologize in advance. Various applications using both R and EViews are given so I suggest you obtain these software and learn to be familiar with them. Given that some topics are difficult, I made accompanying lecture videos on select topics which can be found on my YouTube channel. Merely search for my name, that's Justin Eloriaga, and look for the playlist on Applied Time Series. Feel free to like, share, and subscribe. You are also free to share these lecture notes with your fellow students or colleagues. This will also be useful should you opt to take a comprehensive exam in this course. The contents of these lecture notes are heavily derived from Brooks (2019), Hamilton (1994), and the IMF. While taking this course, I suggest you read on topics in advance if possible and try out the examples for yourself. Trust me, it will all be worth it. Use the force.

Acknowledgement



*"We've passed on all we know. A thousand generations live in you now. But this is your fight" - Luke Skywalker
Star Wars Episode IX: The Rise of Skywalker*

As an academic, we always believe in Isaac Newton's humble belief that we all stand on the shoulders of giants. As such, I would like to express my thanks to the giants who have passed down all they know for me to be able to teach the course as a fellow academic. The author would like to thank Dr. Cesar Rufino, Dr. Lawrence Dacuycuy and Prof. Luisito Abueg for their valuable inputs and lectures which has served as part of the motivation behind this compilation. The conduct of providing lecture notes for students to have an easier time was taken from my academic mentor, Dr. Angelo Unite, who knows that this compilation is certainly up to par with one of his own compilations in the lectures he teaches. I would also like to thank Dr. Marites Tiongco for her valuable insights on how to design the course, that being, a focus on exercises and empirical applications vis-a-vis giving examinations. I believe the students will be eternally grateful.

All contacts of the respective persons mentioned above may be obtained from the De La Salle University School of Economics Website. For DLSU students with a great interest in econometrics, I invite you to join the metrics cluster and join me and fellow metrics professors in shaping the field.

How to Read



*"You know, sometimes, I amaze even myself" - Han Solo
Star Wars Episode IV: A New Hope*

I will by no means tell you how to read, per se. Since you made it this far, this is something authors would care to assume as given. What I will suggest is how to properly go through the lectures. First, I strongly suggest that you read up on your old notes on basic econometrics. I have written the text under the assumption that the reader has some familiarity with basic econometric techniques and constructs. While you do not need to be a *master* in it, I at least need you to be familiar with key assumptions and terms. Secondly, try to read the text and visualize it by writing a few thoughts on the side. Time series is sometimes quite abstract, hence, in the lectures, we will try to visualize this as best as we can. Third, follow the read - video - read technique. It is suggested that you read the notes first, then watch the corresponding video lecture about it, and then reread the notes again. This is so that learnings are reinforced and connections between what was relatively *fluffy* prior to watching the video or the first read isn't anymore. Fourth, please practice doing the codes in R. Central to my objectives for the course is for you to be able to independently code and manipulate data to be able to forecast confidently. Please read the guide, understand the commands, and watch the videos to code confidently. Lastly, take frequent breaks. It is not recommended that you study for the course of an hour straight or more. Try to break it up into chunks to better your understanding and to not intimidate yourself. At the end of this course, I hope you will amaze yourself of the progress you have made. Previous students of mine have been far more confident in handling coding languages and generating reports for their respective workplaces. I trust that you will continue on that legacy!

I also recommend listening to the music of John Williams, the great composer and the Academy award winner. His music will truly make your study time more epic than it actually is. His music is widely available on Apple Music and Spotify. Wouldn't it be nice to listen to the "Imperial March" while deriving a long form proof. That's something truly epic, in my book.

Contents

1	Introduction to Time Series	9
1.1	Time Series Data	9
1.2	Stochastic Process and a Time Series	9
1.3	Weak and Strong Stationarity	10
1.3.1	Strong Stationarity	11
1.3.2	Weak Stationarity	11
1.3.3	White Noise Process	12
1.3.4	Significance of an Autocorrelation Coefficient	12
1.3.5	Box-Pierce Test and the Ljung-Box Test	12
1.4	In-sample and Out of Sample Forecasting	13
2	Univariate Stationary Forecasting Models	13
2.1	Moving Average Model	13
2.1.1	Properties of the Moving Average Model	14
2.1.2	Naïve Forecasting using the Moving Average Model	16
2.2	Autoregressive Model	17
2.2.1	The Stationarity Condition	18
2.2.2	Unconditional Mean of an Autoregressive Model	19
2.2.3	Partial Autocorrelation Function	20
2.2.4	Naïve Forecasting using the Autoregressive Model	20
3	Identification and Selection of a Forecasting Model	21
3.1	Identification through Visual Inspection	22
3.2	Identification through the Autocorrelation Function and the Partial Autocorrelation Function	22
3.2.1	R Example Using a Simulated Series	23
3.3	Model Selection Criteria	26
3.3.1	Fit vs. Parsimony	27
3.3.2	White Noise Errors	28
4	Non-Stationarity	28
4.1	Stochastic Non-Stationarity	28
4.2	Deterministic Non-Stationarity	30
4.3	Consequences of Non-Stationarity	30
4.3.1	Spurious Regression	31
4.4	Integrated Stochastic Process	31
4.5	Testing for Non-Stationarity	31
4.5.1	Dickey-Fuller Test	32
4.5.2	Augmented Dickey-Fuller Test	32
4.5.3	Limitations of the Dickey Fuller and Augmented Dickey Fuller	32
4.5.4	Phillips-Perron Test	33
4.5.5	Kwiatkowski-Phillips-Schmidt-Shin Test	33
5	Forecast Uncertainty and Evaluation	34
5.1	Sources of Forecasting Uncertainty	34
5.1.1	Randomness of an Economic Variable and the Data Generating Process	34
5.1.2	The Value of the Disturbance and the Conditional Mean	34
5.1.3	Using Explanatory Variables	35
5.1.4	Complications in the Parameters and the Conditional Mean	35
5.2	Measures of Forecast Uncertainty	36
5.2.1	Common Statistical Measures for Forecast Uncertainty	36
5.2.2	Bias, Variance, and Covariance Proportion	37
5.2.3	Symmetric Costs	38

5.2.4	Comparative Evaluation	38
5.2.5	Theil's U-Statistics	38
5.3	Forecasting Strategies and Structural Breaks	39
5.3.1	Fixed Window Estimation	39
5.3.2	Expanding Window Estimation	39
5.3.3	Rolling Window Estimation	39
5.3.4	Structural Breaks	40
6	ARMA, ARIMA, and SARIMA	41
6.1	Combining AR and MA	41
6.2	Autoregressive Integrated Moving Average	41
6.3	Seasonal Autoregressive Integrated Moving Average	42
7	Univariate Forecasting Example	42
7.1	Preliminary Steps	42
7.1.1	Installing and Loading Packages	42
7.1.2	Loading the Dataset	43
7.1.3	Declaring a Time Series Object	43
7.1.4	Graphing the Time Series	43
7.1.5	Some Summary Information	44
7.2	Forecasting Building Blocks	44
7.2.1	Generating the ACF and the PACF	44
7.2.2	Decomposition of the Time Series	45
7.3	Tests for Non-stationarity	46
7.3.1	Augmented Dickey Fuller Test	46
7.3.2	Phillips Perron Test	47
7.3.3	Kwiatkowski-Phillips-Schmidt-Shin Test	47
7.4	Forecasting Proper	47
7.4.1	Splitting the Dataset into Training and Testing Sets	47
7.4.2	Diagnosing the Training Set	48
7.4.3	Building the Forecast Model	49
7.4.4	Generating the Forecasts and Forecast Evaluation	50
7.4.5	Showing the Model Gauntlet	51
7.5	Out of Sample Forecasting	53
7.5.1	Generating the Optimal Fit	53
7.5.2	Generating the Out-of-Sample Forecast	53
7.5.3	Some Concluding Thoughts	54
7.6	Code Summary	55
8	Vector Autoregression	59
8.1	The Sim's Critique	59
8.2	Uses of VARs	59
8.3	Mathematical Foundations of the VAR	60
8.4	Stationary VAR	61
8.4.1	Selection of Variables in a VAR	61
8.4.2	Establishing Stationarity	62
8.4.3	Wold's Decomposition Representation	62
8.4.4	Lag Selection Criteria	63
8.5	Forecasting with VAR	64

9	Structural Vector Autoregression	65
9.1	Structural VAR Philosophy	65
9.1.1	Monetary Policy Targeting	65
9.1.2	Fiscal Policy Targeting	65
9.1.3	Isolating Impacts	65
9.2	Identifying a Structural VAR	66
9.2.1	General Overview of Identification and Imposing Restrictions	66
9.2.2	Linking the Structural and Reduced Form VAR	67
9.2.3	Imposing Restrictions	68
9.3	Policy Applications for SVAR	70
9.3.1	Impulse Response Functions	70
9.3.2	Forecast Error Variance Decomposition	70
9.3.3	Granger Causality	70
9.4	Recursive Ordering in SVAR	71
10	Vector Autoregression Example	73
10.1	Preliminaries	73
10.1.1	Initial Steps	73
10.1.2	Testing for Non-Stationarity	74
10.2	Estimation Proper	75
10.2.1	Building the VAR System	75
10.2.2	Lag Selection	75
10.2.3	Estimating the VAR Model	75
10.2.4	Autocorrelation	76
10.2.5	ARCH Test	76
10.2.6	Normality of the Residuals	76
10.2.7	Stability Test	76
10.3	Policy Simulations	77
10.3.1	Granger Causality	77
10.3.2	Impulse Response Functions	77
10.3.3	Forecast Error Variance Decomposition	79
10.3.4	Forecasting using a VAR	79
10.4	Code Summary	80
11	Structural Vector Autoregression Example	83
11.1	Preliminaries	83
11.1.1	Installing and Loading the Required Packages	83
11.1.2	Time Series Loading	83
11.1.3	Plotting the Series	83
11.2	Setting Restrictions and Building the SVAR	84
11.2.1	Setting Restrictions	84
11.2.2	Binding Variables and Selecting the Choleskey Ordering	85
11.2.3	Lag Order Selection	85
11.2.4	Building the SVAR	85
11.3	SVAR Applications	86
11.3.1	Impulse Response Functions in SVAR	86
11.3.2	Forecast Error Variance Decomposition in SVAR	86
11.4	Code Summary	88
12	Vector Error Correction Model	90
12.1	Foundations of Cointegration	90
12.1.1	How a VAR performs under Cointegration	91
12.2	Cointegration and Error Correction	91
12.2.1	Illustrating an Error Correction	92

12.3	VECM Specification	94
12.4	Testing for Cointegration	95
12.4.1	Johansen's Methodology	96
12.5	Johansen Cointegration in R	96
12.5.1	Code Summary	98
13	Course Problem Sets	99
13.1	Problem Set 1 - Univariate Time Series	100
13.2	Problem Set 2 - Multivariate Time Series	101
14	Supplementary Code Appendix	102
15	Course Syllabus	106

1 Introduction to Time Series

"The lightsaber, this is the weapon of a Jedi Knight. Not as clumsy or random like a blaster. An elegant weapon for a more civilized age" - Ben Kenobi

Time series econometrics has sprung up in popularity over the past few years. Anticipating the effects of policies, measuring spillovers of economic actions, and preparing for scenarios have led to a boom in forecasting. Moreover, the economist has needed to add this very skill to their ever growing toolkit. Just as the Jedi relied on their lightsaber, economists must rely on their tools as an extension of the theory to relay it in practice. It is not farfetched that some economists believe their "lightsaber" is econometrics, and for many economic forecasters, their main bane is indeed time series econometrics.

While old Ben Kenobi suggests that the lightsaber is an elegant weapon, I can firmly say that time series is not an elegant subject matter. It is more clumsy (in a sense random) and out there than most econometric fields. However, mastering time series requires both art and science. We shall see that despite the randomness that is time series, we can elegantly derive models that can be used to predict economic variables and control for unobservables. One just needs to trust in the force, which is, in time series, the data generating process and the artful transformations we can do in modelling.

The econometric goals of analyzing economic data, validating economic phenomenon, forecasting economic quantities and controlling for unobserved happenings are clearly seen in the goals of time series econometrics. This course was designed to give the student a rigorous introduction to the world of time series econometrics in an attempt for him or her to wander and explore the many uses of the field. This course has a direct application to the world of microeconomics in understanding firm behaviour, financial market trends and deviations, and consumer decisions. Likewise, this course factors heavily into macroeconomics with applications to forecasting economic quantities, controlling for unobservables, and modelling the economy.

1.1 Time Series Data

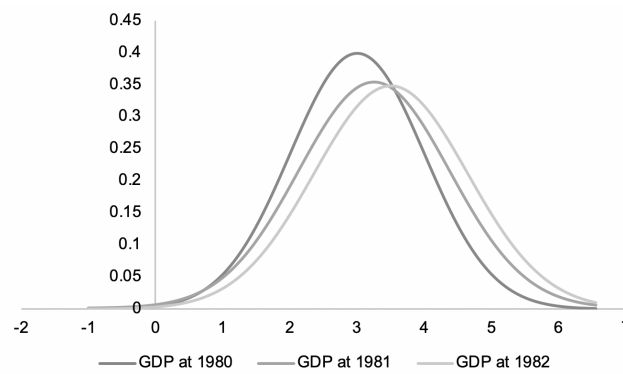
Time Series data which is generally one cross sectional unit with data points in multiple periods of time is generally the most available type of data one could obtain. If you think about just the notion of data, we probably visualize time series data since it's common. Whether it's how we picture the growth of the Philippines overtime (i.e. GDP across years) or how tall we've grown (i.e. Height across years), we always visualize time series data.

While time series may be abundant and in relatively good supply, dealing with time series data can often be a pain at least econometrically. This is because of some rather strict assumptions we place in order for time series to behave properly in an econometric respect. But that shouldn't deter is from tackling it as it is a treasure trove of knowledge and potential understanding not only in finance but for other disciples as well.

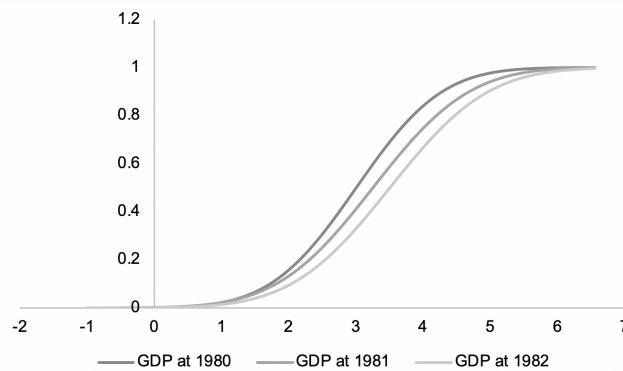
1.2 Stochastic Process and a Time Series

Consider a time series variable like GDP growth. The variable is a heavily tracked variable, especially among economists as it defines how fast (or slow) the productive capacity of the economy is growing. Since the early 1900s, we have used GDP to track the total value of domestic production in the country and as such, developed a relatively long set of values for the series.

The value that GDP could take in a particular time period could be random (ala stochastic). Consider the probability density function and cumulative distribution function in Figure 1. In the figure, we can clearly see that there is a great chance that the growth of the economy will be around 3 percent in the next three years.



(a) Probability Density Function



(b) Cumulative Density Function

Figure 1: PDF and CDF of GDP

The distribution of GDP growth precludes that it could potentially take many values under the distribution with varying odds. We know that it is most likely that the country would grow by around 3 percent (as seen in the center of the distribution) but it is also possible that the country grows by 6.5 percent or post stagnant growth at 0 percent which lie around the tail.

Now consider the distribution that GDP could take not just in one year but every single other year. Assuming its measured yearly, GDP has some distribution each year and it could potentially take some value under a particular distribution each year. The potential values that GDP could take is what we refer to as a *stochastic process*. Now, of course, each and every year, the actual value of GDP is reported and that value of GDP is realized. That realized value is a datapoint in our *time series*. Essentially, a time series is some realization of a stochastic process at a given point in time. Think of it another way. The stochastic process are all the possibilities that GDP could take for a given number of years. The time series is a realized draw from those possibilities.

1.3 Weak and Strong Stationarity

There are generally two classes of processes, the first of which is a *stationary* process and the other is a *non-stationary* process. We will discuss non-stationary processes in the next module, but for now, we zero in on stationary processes.

One of the main considerations we have to make about a certain time series (time series data) is *stationarity*. As said in Brooks (2014), determining the stationarity of a time series will have significant implications on how a series behaves and the various properties the series would have.

There are generally two types of stationarity, the first one is strong stationarity and the second one is weak stationarity, or as some textbooks might call, covariance stationary.

1.3.1 Strong Stationarity

According to Tong (1990) as approached by Brooks (2014), a strictly stationary process is one where for any observation (time) $t_1, t_2, \dots, t_T \in Z$, any specific time say $k \in Z$ and $T = 1, 2, \dots$ that the joint distribution function F of the set of random variables from one period will equal that of any other other period k periods ahead. Mathematically, we denote this as

$$F_{y_{t_1}, y_{t_2}, y_{t_3}, \dots, y_{t_T}}(y_1, \dots, y_T) = F_{y_{t_1+k}, y_{t_2+k}, y_{t_3+k}, \dots, y_{t_T+k}}(y_1, \dots, y_T)$$

To the point, we can consider a series as strictly stationary if the distribution of the values remains the same as time progresses. As such, this implies that the probability that y would fall within a particular interval is the same now as at any time in the future or in the past.

Relating it to our example earlier on GDP. Recall that at each time period, there could be a multitude of possible values that GDP can take. These multitude of values are defined in the PDF of GDP at a period in time. A process is said to be strictly or strongly stationary if the PDF is the same from one time period to another. This is rather restrictive as this assumes that the PDF of GDP does not change over time and that the possible values it can take are always the same set at each time period.

1.3.2 Weak Stationarity

A series is weakly stationary or covariance stationary if it has a constant mean, constant variance and constant autocovariance. Mathematically, this is denoted as

$$E(y_t) = \mu$$

$$E(y_t - \mu)(y_t - \mu) = \sigma^2 < \infty$$

$$E(y_{t_1} - \mu)(y_{t_2} - \mu) = \gamma_{t_2 - t_1} \quad \forall t_1, t_2$$

As the name implies, this is far less restrictive than the strong stationarity as the PDF will be allowed to vary per period for as long as the properties mentioned above still hold.

The first equation pertains to a constant mean while the second equation pertains to a constant variance (i.e. square differences from μ). The third equation pertains to a constant autocovariance. An *autocovariance* is how some y relates to its previous values. For the case of a weakly stationary series, the way that y relates to its previous values should only depend on the difference between t_1 and t_2 so that the covariance between a present and past value is the same as the covariance between any present and past value. This idea or moment condition is summarized as the autocovariance function.

$$E(y_t - E(y_t))(y_{t-s} - E(y_{t-s})) = \gamma_s, \quad s = 0, 1, 2, 3, \dots$$

When $s = 0$, this suggests that the autocovariance at lag zero is obtained which is the autocovariance of y_t with y_t or simply just the variance of y

$$E(y_t - E(y_t))(y_t - E(y_t)) = E(y_t - \mu)(y_t - \mu) = \sigma^2 = \gamma_0$$

The various covariances γ_s are known as autocovariances in a sense that they are the covariances of y with its own previous values.

Central to the analysis of time series data is the concept of autocorrelation. We measure autocorrelation using the autocovariances normalized by dividing by the variance. Hence, an *autocorrelation coefficient* τ_s is given mathematically in the form below.

$$\tau_s = \frac{\gamma_s}{\gamma_0} = \frac{\gamma_s}{\sigma^2}, \quad s = 0, 1, 2, \dots$$

As such, we expect that when $s = 0$ that $\tau_0 = 1$ since the autocovariance when $s = 0$ is merely the variance of y which is the denominator. If we obtain all τ_s and plot them, we obtain the *autocorrelation function*.

1.3.3 White Noise Process

Another time series construct often thrown about is the notion of a *White Noise Process*. We typically place this process on the error term but it's useful to know this in a more general context. In lay man terms, a White Noise Process is any process with no discernable structure. As such, it displays a zero mean, a constant variance and zero autocovariance except at lag zero. This suggests that each observation is uncorrelated with all other values in the sequence. Mathematically, we denote these moment conditions below.

$$E(y_t) = 0$$

$$\text{var}(y_t) = \sigma^2 = \gamma_0$$

$$\gamma_{t-s} = \begin{cases} \sigma^2, & \text{if } t=s \\ 0, & \text{if otherwise} \end{cases}$$

Furthermore, this implies that the value of the autocorrelation function will be 1 at $s = 0$ and will be zero elsewhere.

If we know that y_t is normally distributed, then we can say that the sample autocorrelation coefficients are also approximately normally distributed (Brooks, 2014)

$$\hat{\tau}_s \sim \text{approx}.N\left(0, \frac{1}{T}\right)$$

We denote the sample size by T and the autocorrelation coefficient at a given lag s as τ_s estimated from a given sample.

1.3.4 Significance of an Autocorrelation Coefficient

We can test for the statistical significance of the autocorrelation coefficients by constructing a confidence interval for the estimated autocorrelation coefficients to determine if the autocorrelation coefficient is statistically different from zero.

For a 95% confidence bound, we can compute for the upper and lower bounds using the form below.

$$\left(-1.96 \times \frac{1}{\sqrt{T}}, +1.96 \times \frac{1}{\sqrt{T}}\right)$$

If the autocorrelation coefficient falls outside of this region for a given s , then the null hypothesis that $\hat{\tau}_s = 0$ is rejected. If it does not fall outside of this confidence region, then we fail to reject the null suggesting that the autocorrelation coefficient is not statistically different from zero.

1.3.5 Box-Pierce Test and the Ljung-Box Test

We can test the joint hypothesis that all lags of the autocorrelation coefficients are simultaneously equal to zero using the Q-Statistic developed by Box and Pierce (1970).

This Q-statistic is mathematically given as

$$Q_{BP} = T \sum_{s=1}^m \hat{\tau}_s^2$$

Where T corresponds to the sample size and m corresponds to the chosen (often maximum) lag length. According to Brooks (2014), these coefficients are squared such that the positive and negative coefficients do not cancel out (ala residuals in CLRM). This particular Q-statistic uses a χ^2 distribution with m degrees of freedom.

However, the Box-Pierce test has poor small sample properties which implies that misleading inferences could be had at small samples. As such, Ljung and Box developed a modified test statistic of the original Box-Pierce statistic more commonly known as the Ljung-Box Test.

$$Q_{LB} = T(T+2) \sum_{s=1}^m \frac{\hat{\tau}_s^2}{T-k}$$

1.4 In-sample and Out of Sample Forecasting

It is also important to distinguish an in-sample forecast (ex-post) and an out of sample forecast (ex-ante). Consider a financial manager with data on a firm's quarterly revenue from Q1 2014 to Q4 2016. The goal of the manager is to forecast the next quarter's revenue using some particular model. The manager may first opt to perform an *in-sample forecast* in which he could cut his sample such that he can forecast quarter/s he would have data on presently. Why would he want to do this seemingly redundant task? Well, he would want to see whether or not his forecast model is actually accurate enough to forecast. Likewise, he can use all the data he has to forecast the next few quarters. We refer to this as an *out of sample forecast*. We will illustrate this clearly in the figures to follow.

Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4	Q1	Q2
2014				2015				2016				2017	

Figure 2: In-Sample Forecast

Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4	Q1	Q2
2014				2015				2016				2017	

Figure 3: Out of Sample Forecast

Figure 2 shows an in-sample forecast. In essence, the manager may withhold a part of his sample (in this case Q1-Q4 2016) and choose to forecast that using the remaining data he has (Q1-Q4 for 2014 and 2015). This is used to check for reliability of forecast models and forecast accuracy to avoid misleading inferences. Figure 3 shows an out of sample forecast. This is the typical convention of a forecast in most people's understanding. Essentially, he uses all the available data he has to forecast values he doesn't have.

2 Univariate Stationary Forecasting Models

Forecasting is an essential goal of econometrics. In particular, time series forecasting is a crucial field wherein many stakeholders are researchers lie in anticipation for the next best model. This is because forecasts are useful in generating adequate information as majority of economic and financial decisions require an idea of longer term possibilities which requires a degree of confidence. We have two main ways to forecast. First, we can use past and present values of the variable we want to forecast. Second, we can use the past and present values of an error term to forecast. We will highlight these two approaches clearly in the next section.

2.1 Moving Average Model

The first and fundamental model we will discuss is known as the *moving average model*. Consider an error term u_t which follows a white noise process (as discussed in the last section). Furthermore, we assume that this white noise error term has zero mean and constant variance. In mathematical terms, the $E(u_t) = 0$ and $var(u_t) = \sigma^2 < \infty$.

If we want to explain or forecast some time series variable y_t , one way to do this is using a linear combination of white noise processes (in this case error terms) such that the value of y_t would depend

on present and past values of a white noise error term. Mathematically, we can notate this as the form below.

$$y_t = \mu + u_t + \theta_1 u_{t-1} + \theta_2 u_{t-2} + \theta_3 u_{t-3} + \dots + \theta_q u_{t-q}$$

Note that we often denote a moving average model as $MA(q)$ where q represents the number of moving average lags or the q^{th} moving average order. We can also express this in sigma notation in the form below.

$$y_t = \mu + \sum_{i=1}^q \theta_i u_{t-i} + u_t$$

Central to understanding the moving average model (and the autoregressive model later) is the concept of the *lag operator*. For example, if we want to mathematically represent the immediate past value of y_t , we can denote this in lag operator notation as Ly_t which is equal to y_{t-1} since the exponent of L is 1. Hence, we generalize this as the past value of y_t for i periods ago as $L^i y_t = y_{t-i}$. We can rewrite our past notations as

$$y_t = \mu + \sum_{i=1}^q \theta_i L^i u_t + u_t$$

Alternatively, we can use the Lag Operator notation in a more generalized form as

$$y_t = \mu + \theta(L) u_t; \quad \theta(L) = 1 + \theta_1 L + \theta_2 L^2 + \theta_3 L^3 + \dots + \theta_q L^q$$

2.1.1 Properties of the Moving Average Model

To get to know the properties of the moving average model, consider the example to follow.

Say that we form an $MA(3)$ model which is given below. Note that all u_t follow a white noise process with mean zero and variance σ^2

$$y_t = u_t + \theta_1 u_{t-1} + \theta_2 u_{t-2} + \theta_3 u_{t-3}$$

We first calculate for the mean of y_t . For algebraic simplicity, we will omit the value of the model intercept μ without any loss of generality.

$$E(y_t) = E(u_t + \theta_1 u_{t-1} + \theta_2 u_{t-2} + \theta_3 u_{t-3})$$

Applying properties of expected value

$$E(y_t) = E(u_t) + \theta_1 E(u_{t-1}) + \theta_2 E(u_{t-2}) + \theta_3 E(u_{t-3})$$

But recall that the error terms follow a white noise process that for any u_t , the zero mean property which is $E(u_t) = 0$ holds. Hence, $E(u_{t-i}) = 0$ holds as well. Therefore, the mean of y_t is equal to zero.

$$E(y_t) = 0$$

We then calculate for the variance of y_t . We can calculate for the variance by applying properties of expected value.

$$\text{var}(y_t) = E[y_t - E(y_t)][y_t - E(y_t)]$$

But recall that we determined that $E(y_t) = 0$. Therefore,

$$\text{var}(y_t) = E[y_t][y_t]$$

Replacing y_t with the given model

$$\text{var}(y_t) = E[(u_t + \theta_1 u_{t-1} + \theta_2 u_{t-2} + \theta_3 u_{t-3})(u_t + \theta_1 u_{t-1} + \theta_2 u_{t-2} + \theta_3 u_{t-3})]$$

Applying rules of polynomials

$$\text{var}(y_t) = E[(u_t^2 + \theta_1^2 u_{t-1}^2 + \theta_2^2 u_{t-2}^2 + \theta_3^2 u_{t-3}^2 + \text{cross products})]$$

We use cross products as a lump sum term which represents any linear combination such as $\theta_1 u_{t-1} \theta_2 u_{t-2}$ which could be brought in due to distribution. However, note that the value of this cross product which is $E(u_t, u_{t-i}) = \text{cov}(u_t, u_{t-i}) = 0$ via non-autocorrelation. Moreover, this stems from the property of the autocovariance of a white noise process which suggests that it has zero autocovariance elsewhere except at lag zero.

Hence, the variance of y_t which is also equal to the autocovariance at lag zero is equal to

$$\text{var}(y_t) = \gamma_0 = E(u_t^2 + \theta_1^2 u_{t-1}^2 + \theta_2^2 u_{t-2}^2 + \theta_3^2 u_{t-3}^2)$$

Applying properties of expected value

$$\text{var}(y_t) = \gamma_0 = E(u_t^2) + \theta_1^2 E(u_{t-1}^2) + \theta_2^2 E(u_{t-2}^2) + \theta_3^2 E(u_{t-3}^2)$$

But recall that the variance of the white noise process is $\text{var}(u_t) = \sigma^2$

$$\text{var}(y_t) = \gamma_0 = \sigma^2 + \theta_1^2 \sigma^2 + \theta_2^2 \sigma^2 + \theta_3^2 \sigma^2 = \sigma^2 (1 + \theta_1^2 + \theta_2^2 + \theta_3^2)$$

Let's try to get the autocovariance and subsequently the autocorrelation coefficients for the first four lags.

We compute for the autocovariance for lag 1 using the form below.

$$\gamma_1 = E[y_t - E(y_t)][y_{t-1} - E(y_{t-1})] = E[(y_t)(y_{t-1})]$$

Plugging in y_t and adjusting the model for y_{t-1}

$$\gamma_1 = E[(u_t + \theta_1 u_{t-1} + \theta_2 u_{t-2} + \theta_3 u_{t-3})(u_{t-1} + \theta_1 u_{t-2} + \theta_2 u_{t-3} + \theta_3 u_{t-4})]$$

Again, we apply the properties expected from the cross products

$$\gamma_1 = E(\theta_1 u_{t-1}^2 + \theta_1 \theta_2 u_{t-2}^2 + \theta_2 \theta_3 u_{t-3}^2)$$

Applying properties of expected value

$$\gamma_1 = \theta_1 \sigma^2 + \theta_1 \theta_2 \sigma^2 + \theta_2 \theta_3 \sigma^2 = \sigma^2 (\theta_1 + \theta_1 \theta_2 + \theta_2 \theta_3)$$

We use the same procedure to compute for the autocovariances for lag 2 to 4

For autocovariance at lag 2

$$\gamma_2 = E[(u_t + \theta_1 u_{t-1} + \theta_2 u_{t-2} + \theta_3 u_{t-3})(u_{t-2} + \theta_1 u_{t-3} + \theta_2 u_{t-4} + \theta_3 u_{t-5})]$$

$$\gamma_2 = E(\theta_2 u_{t-2}^2 + \theta_1 \theta_3 u_{t-3}^2)$$

$$\gamma_2 = \theta_2 \sigma^2 + \theta_1 \theta_3 \sigma^2 = \sigma^2 (\theta_2 + \theta_1 \theta_3)$$

For autocovariance at lag 3

$$\gamma_3 = E[(u_t + \theta_1 u_{t-1} + \theta_2 u_{t-2} + \theta_3 u_{t-3})(u_{t-3} + \theta_1 u_{t-4} + \theta_2 u_{t-5} + \theta_3 u_{t-6})]$$

$$\gamma_3 = E(\theta_3 u_{t-3}^2)$$

$$\gamma_3 = \theta_3 \sigma^2$$

For autocovariance at lag 4

$$\gamma_4 = E[(u_t + \theta_1 u_{t-1} + \theta_2 u_{t-2} + \theta_3 u_{t-3})(u_{t-4} + \theta_1 u_{t-5} + \theta_2 u_{t-6} + \theta_3 u_{t-7})]$$

$$\gamma_4 = 0$$

Hence, we can see a definitive property of a moving average model. This property is that all autocovariances for an $MA(3)$ model will be zero for any lag length greater than 3 (say 4).

We can compute for the autocorrelation coefficients for the first 4 lags. Recall that the autocorrelation coefficient $\tau_s = \frac{\gamma_s}{\gamma_0}$

For lag 0

$$\tau_0 = \frac{\gamma_0}{\gamma_0} = 1$$

For lag 1

$$\tau_1 = \frac{\gamma_1}{\gamma_0} = \frac{\sigma^2 (\theta_1 + \theta_1 \theta_2 + \theta_2 \theta_3)}{\sigma^2 (1 + \theta_1^2 + \theta_2^2 + \theta_3^2)} = \frac{\theta_1 + \theta_1 \theta_2 + \theta_2 \theta_3}{1 + \theta_1^2 + \theta_2^2 + \theta_3^2}$$

For lag 2

$$\tau_2 = \frac{\gamma_2}{\gamma_0} = \frac{\sigma^2 (\theta_2 + \theta_1 \theta_3)}{\sigma^2 (1 + \theta_1^2 + \theta_2^2 + \theta_3^2)} = \frac{\theta_2 + \theta_1 \theta_3}{1 + \theta_1^2 + \theta_2^2 + \theta_3^2}$$

For lag 3

$$\tau_3 = \frac{\gamma_3}{\gamma_0} = \frac{\theta_3 \sigma^2}{\sigma^2 (1 + \theta_1^2 + \theta_2^2 + \theta_3^2)} = \frac{\theta_3}{1 + \theta_1^2 + \theta_2^2 + \theta_3^2}$$

For lag 4 (and any lag greater than 3)

$$\tau_s = \frac{\gamma_s}{\gamma_0} = 0 \forall s > 3$$

We can also infer that the Autocorrelation function greater than the specified lag order will also be zero.

2.1.2 Naïve Forecasting using the Moving Average Model

Using the construct of a time series forecast, we can now use the moving average model to try and determine a forecast value for unknown parameters. As we have said, we can either go about the forecast in sample or out of sample. Central to the concept of a forecast is the notion of a conditional probability. We denote a forecast at some time say t as \hat{f}_t which is some forecast wherein we use all data we have on that variable until some time t which we denote as Ω_t . For example, say we want to forecast time $t+1$ using all the data we have today. We can denote this conditional probability as $\hat{f}_{t+1} = E(y_{t+1} | \Omega_t)$.

By definition, a moving average model only has finite memory up to the lag length q and has considerable limitations in producing a sensible forecast horizon. Consider an $MA(4)$ model given below.

$$y_t = \mu + u_t + \theta_1 u_{t-1} + \theta_2 u_{t-2} + \theta_3 u_{t-3} + \theta_4 u_{t-4}$$

We first produce our forecast scalars. This is merely how the model structure would look like at the subsequent forecasts. Say that we want to produce the next 5 forecasts of y_t , we define our forecast scalars as the forms below.

$$y_{t+1} = \mu + u_{t+1} + \theta_1 u_t + \theta_2 u_{t-1} + \theta_3 u_{t-2} + \theta_4 u_{t-3}$$

$$y_{t+2} = \mu + u_{t+2} + \theta_1 u_{t+1} + \theta_2 u_t + \theta_3 u_{t-1} + \theta_4 u_{t-2}$$

$$y_{t+3} = \mu + u_{t+3} + \theta_1 u_{t+2} + \theta_2 u_{t+1} + \theta_3 u_t + \theta_4 u_{t-1}$$

$$y_{t+4} = \mu + u_{t+4} + \theta_1 u_{t+3} + \theta_2 u_{t+2} + \theta_3 u_{t+1} + \theta_4 u_t$$

$$y_{t+5} = \mu + u_{t+4} + \theta_1 u_{t+4} + \theta_2 u_{t+3} + \theta_3 u_{t+2} + \theta_4 u_{t+1}$$

We can now use the conditional probability construct under the assumption that we have to estimate the forecasts for the next 5 periods.

For the first forecast period

$$f_{t+1} = E(y_{t+1}|\Omega_t) = E(\mu + u_{t+1} + \theta_1 u_t + \theta_2 u_{t-1} + \theta_3 u_{t-2} + \theta_4 u_{t-3}|\Omega_t)$$

We now apply properties of expected value. Note that we only have information until time t . Hence, it should be obvious that $E(u_{t+s}|\Omega_t) = 0$. This suggests that the optimal forecast for a zero mean white noise process is just zero.

$$f_{t+1} = E(y_{t+1}|\Omega_t) = \mu + \theta_1 u_t + \theta_2 u_{t-1} + \theta_3 u_{t-2} + \theta_4 u_{t-3}$$

For the second forecast period

$$f_{t+2} = E(y_{t+2}|\Omega_t) = E(\mu + u_{t+2} + \theta_1 u_{t+1} + \theta_2 u_t + \theta_3 u_{t-1} + \theta_4 u_{t-2}|\Omega_t)$$

$$f_{t+2} = E(y_{t+2}|\Omega_t) = \mu + \theta_2 u_t + \theta_3 u_{t-1} + \theta_4 u_{t-2}$$

For the third forecast period

$$f_{t+3} = E(y_{t+3}|\Omega_t) = E(\mu + u_{t+3} + \theta_1 u_{t+2} + \theta_2 u_{t+1} + \theta_3 u_t + \theta_4 u_{t-1}|\Omega_t)$$

$$f_{t+3} = E(y_{t+3}|\Omega_t) = \mu + \theta_3 u_t + \theta_4 u_{t-1}$$

For the fourth forecast period

$$f_{t+4} = E(y_{t+4}|\Omega_t) = E(\mu + u_{t+4} + \theta_1 u_{t+3} + \theta_2 u_{t+2} + \theta_3 u_{t+1} + \theta_4 u_t|\Omega_t)$$

$$f_{t+4} = E(y_{t+4}|\Omega_t) = \mu + \theta_4 u_t$$

For the fifth forecast period

$$f_{t+5} = E(y_{t+5}|\Omega_t) = E(\mu + u_{t+4} + \theta_1 u_{t+4} + \theta_2 u_{t+3} + \theta_3 u_{t+2} + \theta_4 u_{t+1}|\Omega_t)$$

$$f_{t+5} = E(y_{t+5}|\Omega_t) = \mu$$

This clearly shows that our $MA(4)$ model has memory of only up to 4 periods. Anything beyond this (say 5) and the forecast will collapse to the model intercept.

2.2 Autoregressive Model

The second fundamental model is the *autoregressive model*. In hindsight, this model seems like the most intuitive of the forecasting models. If you think about it, it's pretty straightforward. This model uses the past values of y_t to determine its current or future values in addition to an error term (often a white noise error term). This is denoted mathematically in the form below.

$$y_t = \mu + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \phi_3 y_{t-3} + \dots + \phi_p y_{t-p} + u_t$$

We can write this in sigma notation and the lag operator notation which we had discussed in the moving average model section previously

$$y_t = \mu + \sum_{i=1}^p \phi_i y_{t-i} + u_t$$

$$y_t = \mu + \sum_{i=1}^p \phi_i L^i y_t + u_t$$

$$\phi(L) y_t = \mu + u_t; \quad \phi(L) = 1 + \phi_1 L - \phi_2 L^2 - \phi_3 L^3 - \dots - \phi_p L^p$$

2.2.1 The Stationarity Condition

According to Brooks (2014), a desirable property of an AR model is stationarity. This is because if the model is non-stationary, the coefficients of the regression shall have a non-declining effect on the current value of y_t as time progresses.

To test for stationarity, we generally check the roots of the characteristic equation. To explain this, suppose we have a general AR model given below.

$$\phi(L) y_t = u_t; \quad \phi(L) = 1 - L - L^2 - L^3 - \dots - L^p$$

An AR model is stationary if we can rewrite this equation as

$$y_t = \phi(L)^{-1} u_t$$

Where $\phi(L)^{-1}$ is converging to zero. This form above suggests that the autocorrelations will diminish eventually as the lag length is increased. If we assume an infinite lag length, the coefficients ϕ will decline as the lag length increases.

The characteristic equation is given in the form below.

$$1 - \phi_1 z - \phi_2 z^2 - \phi_3 z^3 - \dots - \phi_p z^p = 0$$

Essentially, this characteristic equation is what $\phi(L)$ is equated to zero. By the stationarity condition, it must be that the roots of this characteristic equation should all lie outside the unit circle. These roots determine the characteristics of the process that y_t would take. In particular, the AR process will depend on the roots of this characteristic equation.

Consider the following $AR(1)$ model

$$y_t = y_{t-1} + u_t$$

We can write this in lag order notation as

$$y_t = L y_t + u_t$$

Manipulating and simplifying

$$y_t - L y_t = u_t$$

$$y_t (1 - L) = u_t$$

This suggests that the characteristic equation is

$$1 - z = 0$$

This suggests that $z = 0$ which lies on the unit circle. As such, this particular model is non-stationary. Consider now this $AR(2)$ model given as

$$y_t = 5y_{t-1} - 2.5y_{t-2} + u_t$$

Again, let's express this in lag order notation

$$y_t = 5Ly_t - 2.5L^2y_t + u_t$$

Manipulating and simplifying

$$y_t - 5Ly_t + 2.5L^2y_t = u_t$$

$$(1 - 5L + 2.5L^2)y_t = u_t$$

Therefore, the characteristic equation is

$$1 - 5z + 2.5z^2 = 0$$

We can obtain the roots of this using factorization or the quadratic formula. We obtain that the roots of the characteristic equation are $z = 1.77$ and $z = 0.23$. Since one of the roots is not outside the unit circle, we also conclude that this model is non-stationary.

2.2.2 Unconditional Mean of an Autoregressive Model

Similar with the moving average model, we can identify the properties of an autoregressive model using an example. However, the properties of an AR model are not as straightforward as that of an MA as you will see with this example.

Consider a simple $AR(2)$ model

$$y_t = \mu + \phi_1 y_{t-1} + \phi_2 y_{t-2} + u_t$$

We first calculate for the unconditional mean of y_t .

We use the properties of expected value

$$E(y_t) = E(\mu + \phi_1 y_{t-1} + \phi_2 y_{t-2} + u_t)$$

$$E(y_t) = \mu + \phi_1 E(y_{t-1}) + \phi_2 E(y_{t-2})$$

Notice that we have encountered $E(y_{t-1})$ and $E(y_{t-2})$. Essentially, we can plug in values for this inside the expected value operator following the $AR(2)$ structure.

$$y_{t-1} = \mu + \phi_1 y_{t-2} + \phi_2 y_{t-3}$$

$$y_{t-2} = \mu + \phi_1 y_{t-3} + \phi_2 y_{t-4}$$

Plugging this in to the expected value form earlier

$$E(y_t) = \mu + \phi_1 E(\mu + \phi_1 y_{t-2} + \phi_2 y_{t-3}) + \phi_2 E(\mu + \phi_1 y_{t-3} + \phi_2 y_{t-4})$$

$$E(y_t) = \mu + \phi_1 \mu + \phi_1^2 E(y_{t-2}) + \phi_1 \phi_2 E(y_{t-3}) + \phi_2 \mu + \phi_1 \phi_2 E(y_{t-3}) + \phi_2^2 E(y_{t-4})$$

Simplifying

$$E(y_t) = \mu(1 + \phi_1 + \phi_2) + \phi_1^2 E(y_{t-2}) + \phi_2^2 E(y_{t-4}) + 2\phi_1 \phi_2 E(y_{t-3})$$

It should be clear that we will be approaching an infinite amount of substitutions in this case as we would need to do the same for y_{t+4} and y_{t+3} . There is a pattern emerging which looks akin to the following.

$$E(y_t) = \mu(1 + \phi_1 + \phi_1^2 + \dots + \phi_1^{n-1} + \phi_2 + \phi_2^2 + \dots + \phi_2^{n-1}) + \phi_1^n E(y_{t-n}) + \phi_2^n E(y_{t-n}) + 2\phi_1^n \phi_2^n E(y_{t-n})$$

According to Brooks (2014), for as long as the model is stationary, which is when $|\phi_i| < 1$, then it must mean that $\phi_i^\infty = 0$ as t will perpetually decline as the lag is increased. Hence, by applying simple rules on limits, it follows that $\lim_{n \rightarrow \infty} E(y_{t-i}) = 0$. Therefore, we can get the form below.

$$E(y_t) = \mu (1 + \phi_1 + \phi_1^2 + \dots + \phi_1^{n-1} + \phi_2 + \phi_2^2 + \dots + \phi_2^{n-1})$$

A rule in algebra suggests that the finite sum of an infinite number of geometrically declining terms in a series is just given by the first term in the series divided by 1 minus the common difference.

Therefore, the unconditional mean of y_t is given as

$$E(y_t) = \frac{\mu}{1 + \phi_1 + \phi_2}$$

2.2.3 Partial Autocorrelation Function

The partial autocorrelation function denoted as ψ_{kk} gives the correlation between an observation k periods ago and the current observation after controlling for observations at the immediate lags (i.e. all lags $< k$) (Brooks, 2014).

For example, the PACF of lag 4 would measure the correlation between y_t and y_{t-4} after controlling for the effects of y_{t-1} , y_{t-2} and y_{t-3} .

It follows that at the first lag, the ACF and the PACF are equal since there are no intermediate lag effects to eliminate. Therefore, $\psi_{11} = \tau_1$.

At the second lag, we can determine the partial autocorrelation function using the formula below.

$$\psi_{22} = \frac{(\tau_2 - \tau_1^2)}{(1 - \tau_1^2)}$$

Mathematically, beyond the second lag, the value and the formulas for the subsequent partial autocorrelation coefficients become mathematically rigorous and daunting. However, what we want to emphasize is that there is a connection between the different values of the series at different time periods. As such, there will be non-zero PACFs usually until the lag order of the autoregressive model.

2.2.4 Naïve Forecasting using the Autoregressive Model

Similar to what we have done with the moving average model, we can use an example to illustrate the forecasting procedure underlying an autoregressive model. We will see significant differences as compared to the moving average model.

Consider an $AR(3)$ model given below.

$$y_t = \mu + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \phi_3 y_{t-3} + u_t$$

We first generate our forecast scalars as the parameters are assumed to be stable throughout time. Let's create the first four forecasts.

$$y_{t+1} = \mu + \phi_1 y_t + \phi_2 y_{t-1} + \phi_3 y_{t-2} + u_{t+1}$$

$$y_{t+2} = \mu + \phi_1 y_{t+1} + \phi_2 y_t + \phi_3 y_{t-1} + u_{t+2}$$

$$y_{t+3} = \mu + \phi_1 y_{t+2} + \phi_2 y_{t+1} + \phi_3 y_t + u_{t+3}$$

$$y_{t+4} = \mu + \phi_1 y_{t+3} + \phi_2 y_{t+2} + \phi_3 y_{t+1} + u_{t+4}$$

Similar to the moving average model, we will use the conditional expectation to generate our actual forecasts.

For the first forecast horizon

$$f_{t+1} = E(y_{t+1}|\Omega_t) = E(\mu + \phi_1 y_t + \phi_2 y_{t-1} + \phi_3 y_{t-2} + u_{t+1}|\Omega_t)$$

Applying properties of expectation bearing in mind that we only have information up until time t .

$$f_{t+1} = E(y_{t+1}|\Omega_t) = \mu + \phi_1 y_t + \phi_2 y_{t-1} + \phi_3 y_{t-2}$$

The forecasts of an autoregressive model are unique since an autoregressive model has infinite memory. This merely suggests that we can have forecasts that don't converge to the model intercept (or to zero if it is non-existent) if we exceed the lag order like in the moving average model.

Let's now proceed to the second forecast horizon to see this difference clearly

$$f_{t+2} = E(y_{t+2}|\Omega_t) = E(\mu + \phi_1 y_{t+1} + \phi_2 y_t + \phi_3 y_{t-1} + u_{t+2}|\Omega_t)$$

Applying properties of conditional expectations

$$f_{t+2} = E(y_{t+2}|\Omega_t) = \mu + \phi_1 E(y_{t+1}|\Omega_t) + \phi_2 y_t + \phi_3 y_{t-1}$$

Notice that we have a data point where that point exceeds our data span which is y_{t+1} . However, we do know this conditional expectation which is the forecast from the first forecast horizon. Hence, this second forecast horizon may be written as the form below.

$$f_{t+2} = E(y_{t+2}|\Omega_t) = \mu + \phi_1 f_{t+1} + \phi_2 y_t + \phi_3 y_{t-1}$$

Moving to the third forecast horizon

$$f_{t+3} = E(y_{t+3}|\Omega_t) = E(\mu + \phi_1 y_{t+2} + \phi_2 y_{t+1} + \phi_3 y_t + u_{t+3}|\Omega_t)$$

$$f_{t+3} = E(y_{t+3}|\Omega_t) = \mu + \phi_1 f_{t+2} + \phi_2 f_{t+1} + \phi_3 y_t$$

Moving to the fourth forecast horizon

$$f_{t+4} = E(y_{t+4}|\Omega_t) = E(\mu + \phi_1 y_{t+3} + \phi_2 y_{t+2} + \phi_3 y_{t+1} + u_{t+4}|\Omega_t)$$

$$f_{t+4} = E(y_{t+4}|\Omega_t) = \mu + \phi_1 f_{t+3} + \phi_2 f_{t+2} + \phi_3 f_{t+1}$$

We can now see a defining trend for when we move to further horizons. Hence, the subsequent forecasts may be made in the form below for a horizon s periods away.

$$f_{t+s} = E(y_{t+s}|\Omega_t) = \mu + \phi_1 f_{t+s-1} + \phi_2 f_{t+s-2} + \phi_3 f_{t+s-3}$$

Based on this, can we immediately conclude that the autoregressive model is already better than the moving average model? Not necessarily. While the distinction between finite and infinite memory may be clear in these examples, it does not necessarily explain the accuracy of a forecast model. Those measures will be discussed when we get to the next section.

3 Identification and Selection of a Forecasting Model

Crucial to forecasting is of course using the most adept model in estimating future quantities. While it is certainly feasible to use an iteration of every model, this approach is cumbersome and may cloud the judgement of most forecasters. As such, we need to have tractable measures as to which model would be preferred over others.

In the last section, we talked about the autoregressive model and the moving average model. These models display certain properties which go to to guide us in identifying some things in a time series. There are certain patterns that allow us to associate some models with a certain series. Knowing these patterns and properties allow us to build a tractable framework by which we can judge models and compare them

against others to see which we can use to forecast. In this section, we will try to understand how to identify these patterns through a more visual approach first, and then select among an alternative set of models given various criteria.

3.1 Identification through Visual Inspection

One way we can start is just by visual inspection. As we said earlier, the AR and MA models are stationary models which assume covariance stationarity. To determine if a series is covariance stationary, we can use a number of tactics. One simple tactic is just through visual inspection.

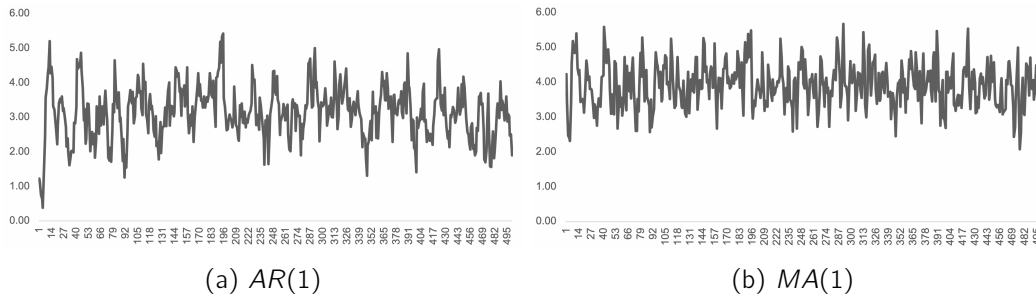


Figure 4: Showing an $AR(1)$ and an $MA(1)$ Process

In Figure 4, we can see an example of an $AR(1)$ and an $MA(1)$ process. To be more specific, it is an $AR(1)$ where the model intercept $\mu = 1$ and $\phi_1 = 0.7$. Ergo, the model may be written as $y_t = 1 + 0.7y_{t-1} + u_t$. Notice that the mean of the series is constant. This is determined graphically if the line of best fit is a flat line. By visual inspection, if the one line that can feasibly connect all points is drawn down the middle, then the series has a constant mean. The series has a constant variance if there are no "wild" shocks that are there. There are hardly any extreme outliers or one off observations. As such, it's fairly safe to say that there's a constant variance. On the other hand, the $MA(1)$ process has a model intercept $\mu = 4$ and $\theta_1 = 0.7$. Ergo, the model may be written as $y_t = 4 + 0.7u_{t-1} + u_t$. The two series look very much the same (barring the intercept which in the $AR(1)$ starts at 1 and at the $MA(1)$ starts at 4). As such, to determine which series is which, we can look into the ACF and the PACF.

3.2 Identification through the Autocorrelation Function and the Partial Autocorrelation Function

In the last subsection, we explored identification using visual inspection. We concluded that while identifying a stationary series is straightforward, determining which exact stationary process it is is quite difficult. This problem can be alleviated once we get into looking at the *autocorrelation function* (ACF) and the *partial autocorrelation function* (PACF).

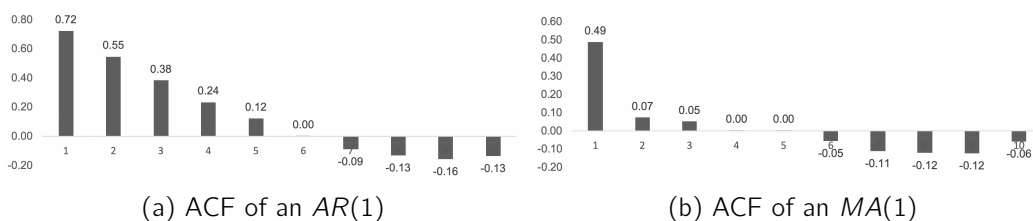


Figure 5: Showing the Autocorrelation Function

In Figure 5, we see the ACF graphs of both an $AR(1)$ process and an $MA(1)$ process. The ACF graph contains the autocorrelation coefficients from the first lag until the 10th lag. Here, we can now see some patterns emerging. For an $AR(1)$ process, there is a *geometric decay*. If you look at the autocorrelation coefficient, you can see that it gradually decays from lag 1 to lag 2, lag 2 to lag 3, and so on. Generalizing for any $AR(p)$, we can see the same trend with some *oscillation* possible. In general, for an $AR(p)$, the ACF decays to zero. If however the $\phi < 0$, observed oscillation will occur. For an $MA(1)$ model, there is

a cutoff after lag 1. Notice the sharp drop in the autocorrelation coefficient's value from lag 1 to lag 2. Generalizing for an $MA(q)$ model, the cutoff occurs at the specified q^{th} lag. Say we had an $MA(5)$, we should expect the abrupt cut-off after lag 5.

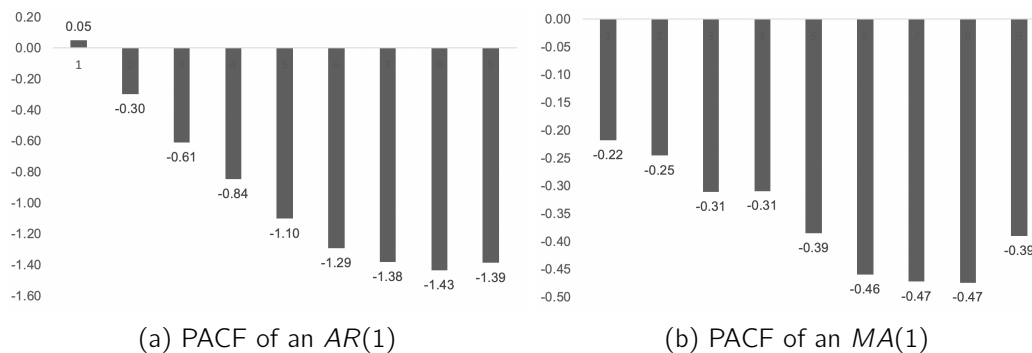


Figure 6: Showing the Partial Autocorrelation Function

Figure 6 shows the PACF of an $AR(1)$ and an $MA(1)$. Observant readers can point out that the behavior of each model is reversed in the PACF. For an $AR(1)$ model, we see the immediate cutoff after lag 1. Generalizing to a higher lag order $AR(p)$ would place the cutoff right after the p^{th} lag. Conversely, for an $MA(1)$ process, we see geometric decay. Again, observed oscillation is expected if $\theta < 0$. Generalizing to a higher order $MA(q)$, we can see the same geometric decay with some oscillation probable.

Table 1: Summary of ACF and PACF Patterns

Model	ACF	PACF
$AR(1)$	Geometric Decay	Cutoff after Lag 1
$AR(p)$	Geometric Decay	Cutoff after Lag p
$MA(1)$	Cutoff after Lag 1	Geometric Decay
$MA(q)$	Cutoff after Lag q	Geometric Decay
$ARMA(p, q)$	Geometric Decay	Geometric Decay

These patterns are summarized in the table above. The pattern expected for an $ARMA(p, q)$ model is also stated. We will explore more into this model in the upcoming sections.

3.2.1 R Example Using a Simulated Series

As an optional activity, you may opt to do this simple R example. To show this in greater detail, let us simulate an AR, MA, and an ARMA process in R. Follow along the codes. A summary code is given in the end as well. First, we need to install the required packages using the `install.packages()` command and call them for use using the `library()` command.

```
install.packages("tseries")
install.packages("forecast")
install.packages("tidyverse")
```

```
library(tseries)
library(forecast)
library(tidyverse)
```

After this, we now simulate the data using the `list` command and the `arima.sim()` command.


```
AR1 <- list(order = c(1,0,0), ar = 0.5, sd = 0.01)
AR1 <- arima.sim(n = 10000, model = AR1)

MA1 <- list(order = c(0,0,1), ma = 0.7, sd = 0.01)
MA1 <- arima.sim(n = 10000, model = MA1)

ARMA11 <- list(order = c(1,0,1), ar = 0.5, ma = 0.5, sd = 0.01)
ARMA11 <- arima.sim(n = 10000, model = ARMA11)
```

As you can see, we created three time series, an $AR(1)$, an $MA(1)$ and an $ARMA(1,1)$. For each model, we simulated using the `arima.sim()` command 10,000 time periods. In the $AR(1)$ we set $\phi = 0.5$, for the $MA(1)$, we set $\theta = 0.7$, and for the $ARMA(1,1)$, we set $\phi, \theta = 0.5$. By now, you should note that we are creating stationary processes. We created the objects `AR1`, `MA1`, and `ARMA11` to represent these series' respectively.

```
combo <- cbind(AR1,MA1,ARMA11)
autoplot(combo, facets = TRUE) + ggtitle("Time Series Plots")
```

We can then graph the said series' to visualize them clearly. As we noted earlier, it's harder to see the differences across the different series just through graphical inspection. As such, we need to visualize their ACFs and PACFs to get a clearer idea of what they are.

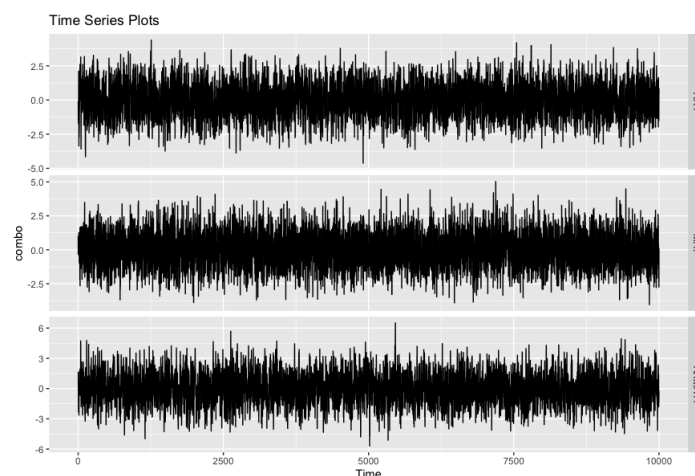


Figure 7: Graphing the Simulated Series in R

We can now derive the ACF and the PACF of the different processes. To do this, we are going to use the `acf()` command and the `pacf()` command. This will highlight the properties and patterns we alluded to earlier. For reference, the `#` key is the hashtag symbol and is used to comment in R code.

```

#For the AR
acf(AR1) # Geometric Decay
pacf(AR1) # Cutoff after first lag

#For the MA
acf(MA1) # Cutoff after first lag
pacf(MA1) # Geometric Decay

#For the ARMA
acf(ARMA11) # Geometric Decay
pacf(ARMA11) # Geometric Decay

```

Generating all of these in one graph will yield us Figure 8. This shows us side-by-side results of the ACF and PACF of each series.

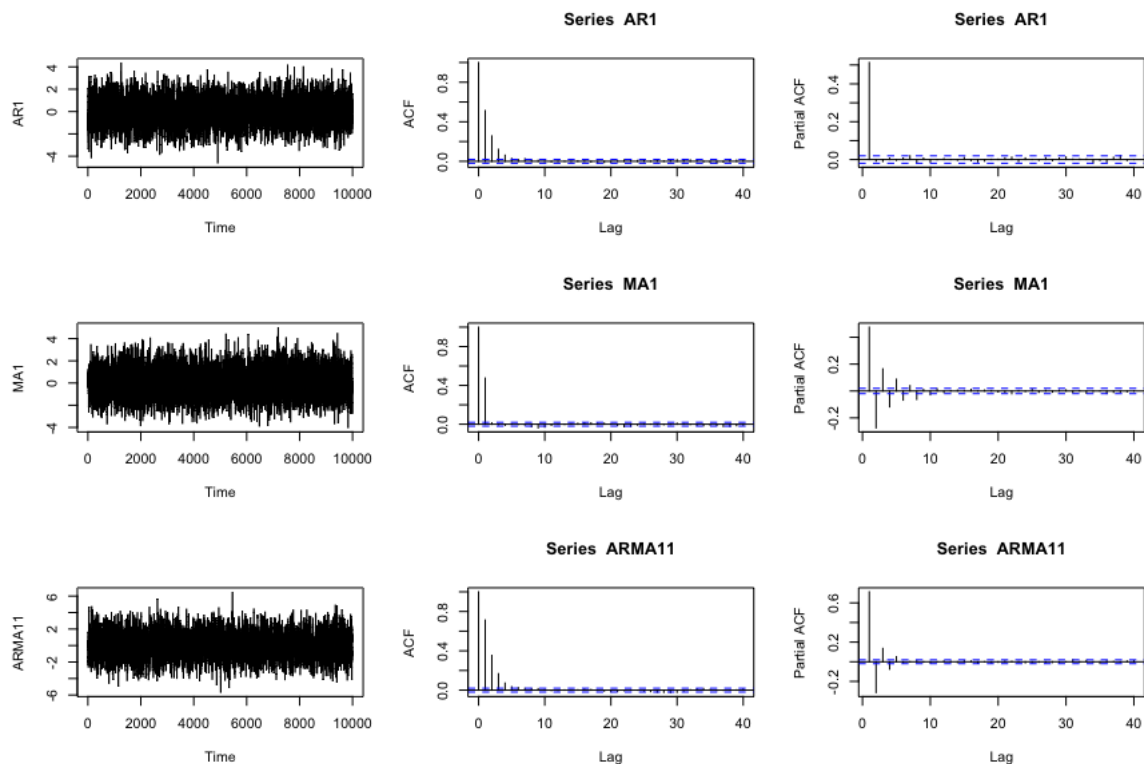


Figure 8: ACF and PACF of the Simulated Series

We can clearly tell that in an $AR(1)$, the ACF is geometrically decaying and the PACF has an immediate cutoff after lag 1. This also suggests that the number of lags to use is 1 (which is correct since the series is an $AR(1)$). Conversely, we see that the ACF of an $MA(1)$ shows the immediate cutoff at lag 1. All other lags are not significant. This suggests that the number of lags to use is 1 lag, which is again correct. Moreover, the PACF of the $MA(1)$ is geometrically decaying/oscillating. Lastly, a geometric decay/oscillation can be seen for an $ARMA(1, 1)$. This can be further generalized to whatever p or q or a combination of both (i.e. in an ARMA) is used (see the comparison table). The summary code follows.

```

# Comparing the ACF and PACF of an AR, MA, and ARMA Processes
# Justin S. Eloriaga

# Installing the Required Packages and Calling them for Use

install.packages("tseries")
install.packages("forecast")
install.packages("tidyverse")

library(tseries)
library(forecast)
library(tidyverse)

# Simulating the Series
AR1 <- list(order = c(1,0,0), ar = 0.5, sd = 0.01)
AR1 <- arima.sim(n = 10000, model = AR1)

MA1 <- list(order = c(0,0,1), ma = 0.7, sd = 0.01)
MA1 <- arima.sim(n = 10000, model = MA1)

ARMA11 <- list(order = c(1,0,1), ar = 0.5, ma = 0.5, sd = 0.01)
ARMA11 <- arima.sim(n = 10000, model = ARMA11)

# Graphing the Three Series

combo <- cbind(AR1,MA1,ARMA11)
autoplot(combo, facets = TRUE) +ggtitle("Time Series Plots")

# Showing the PACF and ACF

par(mfrow = c(3,3))
plot.ts(AR1)
acf(AR1) # Geometric Decay
pacf(AR1) # Cutoff after first lag
plot.ts(MA1) acf(MA1) # Cutoff after first lag
pacf(MA1) # Geometric Decay
plot(ARMA11)
acf(ARMA11) # Geometric Decay
pacf(ARMA11) # Geometric Decay

```

3.3 Model Selection Criteria

We now get to the part of choosing which model we think is best. Choosing the best model is not something straightforward. We need to consider certain tradeoffs and certain characteristics for some given condition. Fortunately, there are certain criteria we can use to help guide our approach. In selecting which model to use, we consider the following:

- Significance of the Coefficients
- Fit vs. Parsimony
- White Noise Residuals
- Ability to Forecast
- Account for Structural Breaks

3.3.1 Fit vs. Parsimony

The first consideration lies in the tradeoff between fit and parsimony. The first aspect, *fit*, is how well your model explains the variation in y_t in such a way that it can be used for forecasting its future values. One way forecasters have done this in the past is to just throw everything in the proverbial kitchen sink. That is, if we want to forecast y_t using an *AR* model, we just use all the lags we have. This is because, as we know, increasing the number of regressors will undoubtedly increase fit (i.e. R^2) as goodness of fit measures are monotonically increasing with the number of regressors in the model. However, merely increasing the number of regressors is prone to many biases. For one, including lags that are irrelevant would have disastrous consequences come forecasting. As such, it is of paramount importance to determine the most adequate lag length. This begs the question, how many lags will we include? We want to include the lags that explain the variation in y_t but we don't want to include all such that the *parsimony* of the model is compromised.

Fortunately, there are a couple of information criteria which we can use to determine the right lag length for us. The first of this information criteria is known as the *Akaike Information Criteria* or AIC for short. The AIC is computed for in the form below.

$$AIC = T \ln(RSS) + 2(p + q + 1)$$

The formula for the AIC may look awkward at first but it is actually fairly intuitive. The first part of the formula contains the *RSS* or the residual sum of squares. The *RSS* pertains to (in the ANOVA framework) the proportion in the total variation of y_t which was left unexplained by the regressors. If the fit of the model is better, then the tendency is that *RSS* decreases. Recall the ANOVA identity that $TSS = RSS + ESS$. If the fit is greater, then *ESS* must have increased and in order to maintain equality, *RSS* must have decreased. Thus, the value of the AIC is smaller when the fit is better. The second term pertains to the number of p and q lags used. Notice that if the value of either p or q increases, then the AIC increases as well. This term is essentially a penalty term for including more lags and preserves the parsimony of the model. This means that the AIC can be broken down into a fit penalty and a parsimony penalty, as shown below.

$$AIC = \underbrace{T \ln(RSS)}_{\text{Fit Penalty}} + \underbrace{2(p + q + 1)}_{\text{Parsimony Penalty}}$$

There are other information criteria that are out there. There is also the Schwarz's Bayesian Information Criterion (SBIC), and the Hannan-Quinn Information Criterion (HQIC).

$$SBIC = T \ln(RSS) + (p + q + 1) \ln(T)$$

$$HQIC = T \ln(RSS) + 2(p + q + 1) \ln(\ln(T))$$

You will notice that the SBIC is a much stiffer penalty than the AIC and that the HQIC lies somewhere in between. The penalty of the SBIC and HQIC are relatively more anchored on parsimony vis-a-vis fit.

3.3.2 White Noise Errors

We talked about what a white noise error term is in the second module. To recall, this is some process that has no definitive structure and has the property of a constant mean, constant variance, and constant autocovariance. Our aim is to eliminate any autocorrelation in the residuals. If there was autocorrelation in the residuals, it could indicate that the model is not specified well enough to reflect the lag structure well.

Typically, what we do first is graph the standardized residuals which is essentially the raw u_t divided by the standard deviation. Moreover, what we do is graph out the ACF and the PACF of the residuals and see if all coefficients would lie inside the confidence bound at a given confidence level. We would then try to see if the residuals have the properties of the white noise process through visual inspection and by looking at the ACF or PACF.

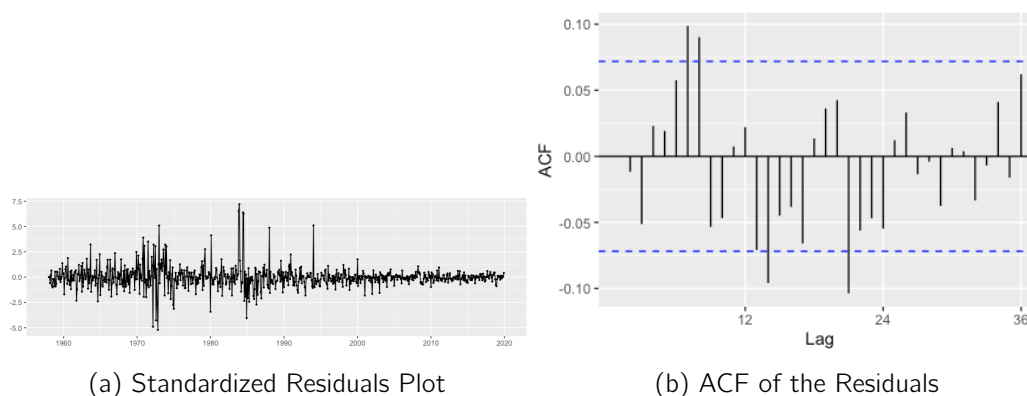


Figure 9: Checking for White Noise Residuals

Figure 9 shows the standardized residual plot and the ACF of the residuals. Notice that the process overall looks stationary. But we do note that there are spikes around certain years. Those spikes cause some autocorrelation coefficients to be significant (beyond the blue confidence interval) suggesting that the residuals are not fully white noise. Ergo, the model specified for this particular forecast was not necessarily the best one chosen or the right model to have used. Similarly, to determine if the residuals of the white noise process, we can also use the Box-Pierce or the Ljung-Box Q statistics defined earlier.

4 Non-Stationarity

Up to this point, we have been dealing with stationary processes. However, much of the time series data are non-stationary. Our assumption of weak stationarity is very restrictive. If that assumption does not hold, then the series is said to be non-stationary. In this section, we will discuss some examples of non-stationary processes, identify the consequences of using such data, and some tests to detect non-stationarity.

In a non-stationary series, there is no long-run mean to which the series returns to. In a sense, there is no 'steady state'. This is crucial from an economic aspect as we believe that the long run equilibrium concept may exist in some variables. Moreover, the variable is time dependent if it is non-stationary. Therefore, essentially, it could go onto infinity. As a result, theoretical autocorrelations don't decay.

4.1 Stochastic Non-Stationarity

A process can have a random trend which varies over time. The classical example of stochastic non-stationary time series is a simple *random walk*.

$$y_t = y_{t-1} + u_t$$

Observant readers will be quick to point out that a random walk is essentially an $AR(1)$. This is true. The random walk is an $AR(1)$ where the lag coefficient of y_{t-1} is equal to 1. That is, $\phi_1 = 1$. Moreover,

the simple random walk can be extended as a random walk *with drift*.

$$y_t = \mu + y_{t-1} + u_t$$

Recall that in any $AR(1)$, the ϕ_1 is the coefficient of the first lagged value. The case where $\phi_1 > 1$ is not typically encountered in economics as the function becomes explosive. However, $\phi_1 = 1$ characterizes a lot of time series.

Having a non-stationary time series and forecasting using the raw series has a lot of consequences. One consequence is that shocks don't decay, but are persistent through time. Moreover, not only are they persistent, they are propagated so that a given shock has an increasingly larger effect or influence, which is highly improbable.

Consider an $AR(1)$ with no drift

$$y_t = \phi y_{t-1} + u_t$$

Let us lag the value of y_t by two periods

$$y_{t-1} = \phi y_{t-2} + u_{t-1}$$

$$y_{t-2} = \phi y_{t-3} + u_{t-2}$$

We can substitute $y_{t-1} = \phi y_{t-2} + u_{t-1}$ to the original $AR(1)$

$$y_t = \phi(\phi y_{t-2} + u_{t-1}) + u_t$$

$$y_t = \phi^2 y_{t-2} + \phi u_{t-1} + u_t$$

Substituting $y_{t-2} = \phi y_{t-3} + u_{t-2}$ to the reformulated $AR(1)$

$$y_t = \phi^2(\phi y_{t-3} + u_{t-2}) + \phi u_{t-1} + u_t$$

$$y_t = \phi^3 y_{t-3} + \phi^2 u_{t-2} + \phi u_{t-1} + u_t$$

I hope you started seeing the trend that this general sequence with recursive substitutions will lead to the following term:

$$y_t = \phi^{T+1} y_{t-(T+1)} + u_t + \phi^1 u_{t-1} + \phi^2 u_{t-2} + \phi^3 u_{t-3} + \dots + \phi^T u_{t-T}$$

Essentially, one way to look at the present value of y which is y_t is the association between it and its very first value and the contributions of successive errors or disturbances.

There are three general cases that we can deduce from this.

1. If $\phi < 1$, this suggests that subsequent $\phi^t \rightarrow 0$ as the number of periods away $T \rightarrow \infty$. In this case, the effect of a shock diminishes as time elapses.
2. If $\phi = 1$, then it suggests that $\phi^t = 1 \forall t$. This suggests that the effects persist, shocks have equal weights and the variance grows indefinitely. In essence, the current value of y is merely an infinite sum of past shocks plus some starting value of y_0 . We call this the *unit root case*, for the root of the characteristic equation (see previous section) would be unity.
3. If $\phi > 1$, shocks become more influential over time. Ergo, this is the explosive case.

Let us now turn our attention to the random walk with drift. We know that our random walk with drift is a non-stationary series and is given by the form below.

$$y_t = \mu + y_{t-1} + u_t$$

In the random walk with drift, there is some stochastic trend in the data. One way we can alleviate the non-stationary nature of the series is through a process known as *differencing*. Essentially, we subtract the present value of y by its immediate past value y_{t-1} .

$$\Delta y_t = y_t - y_{t-1}$$

Manipulating the random walk with drift

$$y_t - y_{t-1} = \mu + u_t$$

$$(1 - L)y_t = \mu + u_t$$

$$\Delta y_t = \mu + u_t$$

This new variable Δy_t is a stationary series. Moreover, the stationarity has been induced when differencing had taken place. Readers should note that the representation $(1 - L)y_t = \mu + u_t$ is essentially a *unit root process*, that is, the root of the characteristic equation

4.2 Deterministic Non-Stationarity

There is a chance that a series is some non-random function of time, that is to say, *deterministic*. The most common of these is a *trend-stationary process*. A trend stationary process typically looks like the form below.

$$y_t = \alpha + \beta t + u_t$$

Note that we assume that u_t is a white noise process. For example, consider the AR(1) with trend model below.

$$y_t = \alpha + \beta t + 0.08y_{t-1} + u_t$$

Essentially, the process would be your typical stationary AR(1) process since $\phi = 0.08 < 1$ except that we have some linear trend βt . To be able to stationarize a series like this, we would need to de-trend (or difference).

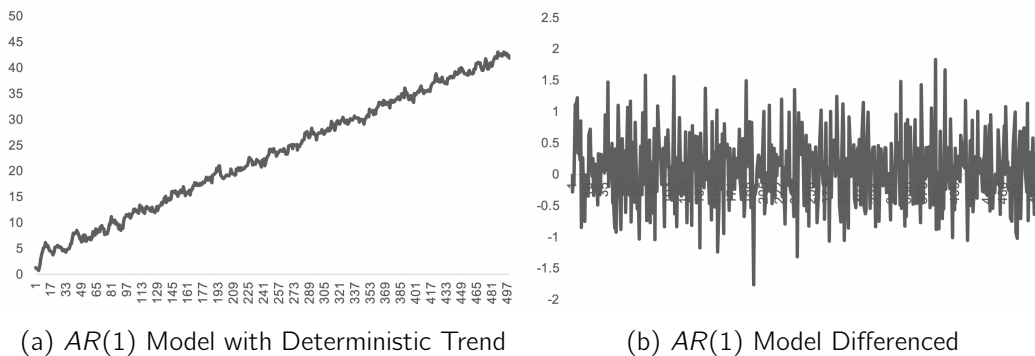


Figure 10: Showing a Trend Stationary Process

In Figure 10, we see the graph of the AR(1) with deterministic trend and the differenced value. Although trend-stationary and difference-stationary are both "trending" over time, the correct approach needs to be used in each case (Brooks, 2014). In the case of an AR(1) with deterministic trend, merely differencing would remove the non-stationarity but would introduce an MA(1) in the trend-stationary model.

4.3 Consequences of Non-Stationarity

There will of course be consequences if we just applied a stationary model to a non-stationary series. The first major reason is that shocks don't die out. We alluded to this earlier in that the effect of the shock might fall under three cases depending on the value of ϕ in the case of an AR(p).

On the otherhand, there will be statistical consequences as well. Firstly, there is a tendency that a bias would exist on the AR coefficients. If a series is non-stationary, then the most logical vantage point is to use the past values rather than the disturbances as if a trend (whether stochastic or deterministic) persists, convention and logic dictates that the past value is the best way to explain the current value. This may lead to us mistakenly estimating an $AR(p)$ model which may lead to a deficient forecast especially when the AR structure isn't fit for that data once it is stationarized. Secondly, there is a non-normal distribution of the test statistics. This may make the t-stats inconsistent or misleading. Third, and most consequential, would be a *spurious regression*.

4.3.1 Spurious Regression

A *Spurious Regression* is a non-sensical regression wherein two unrelated non-stationary series might appear to be related through a standard OLS and produce high goodness of fit measures but are in fact, unrelated.

Consider two unrelated time series, say y_t and z_t . If we ran an OLS model between them, there is a chance that we can conclude that they have a high degree of correlation.

$$y_t = \alpha + \beta_t z_t + u_t$$

The fit of the model may indicate that the two series are related. Indicators like a high R^2 and significant t stats could indicate that everything is good, but this might be misleading.

The true test to determine a good fit would be the stationarity of the regression coefficients. If this stationarity condition isn't met, then the regression is likely a spurious one. Detecting spuriousness is slightly difficult but there are a few indicators that can be used. One of these indicators is the *Granger-Newbold Rule of Thumb*. According to the rule, if the model R^2 is greater than the Durbin-Watson test statistic d , then the regression is likely spurious.

4.4 Integrated Stochastic Process

Given the consequences of non-stationary processes, there is a need to stationarize them. As we had mentioned, one way to stationarize a series is through the use of *differencing*.

Consider a non-stationary series y_t which must be differenced d number of times in order to be stationary. In that case, it is said that the series y_t is "integrated" of order d . More formally, this is writted as $y_t \sim I(d)$. Essentially, d is the degree of integration. For example, say we have a non-stationary variable $z_t \sim I(1)$. Since z_t 's order of integration is 1, it must be differenced once to become stationary.

4.5 Testing for Non-Stationarity

Now that we have some idea of the consequences of non-stationarity as well as how to deal with it. We must now turn our attention to testing for the presence of non-stationarity. While we can always graph the function and identify through visual inspection, there is a great chance of mis-identification, especially in the case of trend stationary series'. Central to the construct of testing for non-stationarity is the concept of a *unit root*. We touched briefly on this concept before but let us walk through it more intuitively.

Consider an $AR(1)$ process

$$y_t = \phi y_{t-1} + u_t$$

We said before that stationarity requires that $\phi < 1$. More specifically, the absolute value of $|\phi| < 1$. Recall that if $\phi = 1$, we end up with a random walk which is non-stationary. When $\phi > 1$, we end up with an explosive process. In practice, the mosy common occurence of non-stationarity in economics is that $\phi < 1$ or $\phi = 1$. In the latter, the series is non-stationary. Recall the concept of the characteristic equation discussed earlier in the stationarity condition of an autoregressive process.

$$1 - \phi_1 z - \phi_2 z^2 - \phi_3 z^3 - \dots - \phi_p z^p = 0$$

Essentially, this characteristic equation is what $\phi(L)$ is equated to zero. By the stationarity condition, it must be that the roots of this characteristic equation polynomial should all lie outside the unit circle or should be greater than 1. If one of the roots is equal to 1, then y_t is said to have a unit root.

4.5.1 Dickey-Fuller Test

One of the most pioneering tests for detecting non-stationarity was the Dickey Fuller Test based on Fuller (1976) and Dickey and Fuller (1979). Essentially, the test examines the value of ϕ . In particular, it tests the null hypothesis that $\phi = 1$ against the alternative that $\phi < 1$. In practice, the test implores the use of the differenced form.

$$\Delta y_t = \psi y_{t-1} + u_t$$

We derive this by using the first $AR(1)$ and it's immediate lag. If we subtract the immediate lag of the y_t , i.e. y_{t-1} to the both sides of the equation, we are essentially getting this difference equation. In this case, $\psi = \phi - 1$. The reason why the difference case is used is for ease of computation and interpretation. Essentially, we are regressing Δy_t on its lag and testing for the significance of the coefficient.

Likewise, the test can be extended further to accomodate the inclusion of an intercept and a deterministic time trend.

$$\Delta y_t = \psi y_{t-1} + \mu + \beta t + u_t$$

As with the base difference equation, the null and alternative hypothesis are formulated in the manner

$$H_0 : \psi = 0$$

$$H_a : \psi \neq 0$$

If the value of $\psi = 0$, then it must be that $\phi = 1$ which we know precludes a non-stationary series. As such, a rejection of the null hypothesis suggests that a series is stationary.

4.5.2 Augmented Dickey-Fuller Test

So far, the Dickey Fuller test assumes that u_t is a white noise error term. However, if u_t is autocorrelated, we would need a drift version of the test which allows for higher order lags. Running the original Dickey Full test in this case would result in an oversized test suggesting that the true size of the test which is the proportion of times a correct null hypothesis is incorrectly rejected would be higher than the normal sized used. As such, we 'augment' the test using p lags of the original series. This is known as the *Augmented Dickey Fuller Test*

$$\Delta y_t = \psi y_{t-1} + \mu + \alpha t + \sum_{i=1}^p \beta_i \Delta y_{t-i} + u_t$$

Adding the lags of Δy_t would account for the dynamic structure present in y_t and ensures that u_t is not autocorrelated. Like before, we conduct the test of ψ following the same hypotheses.

4.5.3 Limitations of the Dickey Fuller and Augmented Dickey Fuller

The use of these tests forces us to choose. It forces us to choose the optimal number of lags of the dependent variable or the forecast variable in question. While we can use the AIC, SBIC, and HQIC, we can also use a General to Specific Approach (i.e. Trial and Error). Typically, the most preferred is the AIC and the SBIC and HQIC penalize relatively heavily on parsimony. However, we find that the test statistic of these tests are very sensitive to changes in the lag structure. Moreover, the functional form of the

model must also be specified. Do we include a trend, is there an intercept, do we use both a time trend and an intercept? These are functional specifications we have to consider before running the test.

The Dickey Fuller and the Augmented Dickey fuller have been found to have low power (i.e. power to reject the null hypothesis) in some circumstances. Consider a model wherein $\phi = 0.95$. By all accounts, it meets our criteria for a stationary process but the result of the test may indicate non-stationarity especially in data with a low sample size. Moreover, these tests have not been really reliable when dealing with trend stationary processes.

The need for a more generalized test has been met by the Phillips Perron Test and the KPSS test, however, there is no one best test to consider the best. As such, better to perform all tests if possible.

4.5.4 Phillips-Perron Test

The Phillips-Perron (PP) test is also a unit root test. However, this test has a few desirable properties to which the standard Dickey Fuller Test and the Augmented Dickey Fuller test cannot lay claim to. First, the PP Test corrects for any serial correlation and heteroscedasticity in the errors by some direct modification to the test statistics. Second, there is no need to specify the lag length. The formulation of the test uses the differenced equation to follow.

$$\Delta y_t = \psi y_t + \mu^* + \delta t + u_t, \quad u_t \sim I(0), ARMA(p, q)$$

The equation is similar to before. The modification lies in the distribution of u_t which is stationary at level ($I(0)$) but may follow an $ARMA(p, q)$ process. Hence, the error is not assumed to be white noise. It can take a purely AR process, MA process or a combination of the two.

$$H_0 : \psi = 0$$

$$H_a : \psi \neq 0$$

As this is a unit root test, we still test the coefficient ψ to test for non-stationarity. Hence, if the series is non-stationary, we fail to reject the null hypothesis.

4.5.5 Kwiatkowski-Phillips-Schmidt-Shin Test

This test, whose name is certainly a mouthfull, is the KPSS test and it is a stationarity test. The KPSS test doesn't test for the presence of unit roots, rather, it determines whether a series is stationary or non-stationary using a different procedure. Recall that one of the main limitations of the DF and ADF tests would be detecting trend stationary processes. This is a relative speciality of the KPSS test.

The formulation of the model hinges on the null hypothesis that y_t is trend stationary.

$$y_t = \beta_0 D_t + \mu_t + u_t, \quad u_t \sim WN$$

$$\mu_t = \mu_{t-1} + \varepsilon_t, \quad \varepsilon_t \sim N(0, 1)$$

In the model above, D_t contains deterministic components. These components may be a constant or a constant plus time trend. In the KPSS test, we test the variance of the error term of the equation on μ_t . The null and alternative hypothesis are specified in this manner.

$$H_0 : \sigma_\varepsilon^2 = 0$$

$$H_a : \sigma_\varepsilon^2 > 0$$

If it was found that $\sigma_\varepsilon^2 = 0$, it means that μ_t is just a constant and reduces to a trend μ . Therefore, y_t is trend stationary. If however, the variance is significantly different from zero, then μ varies over time suggesting that y_t is not stationary.

5 Forecast Uncertainty and Evaluation

"Never tell me the odds" - Han Solo

The odds of creating a "magic bullet" model to forecast perfectly are about the same as me being able to purchase an actual Millenium Falcon in my lifetime.

While identification and estimation underlie the heart of forecasting, controlling and evaluating forecasts are just as important. In this section, we will learn how to choose between a slew of competing models and forecasts to determine which is best. We will also find a way to assign some mathematical value to judge the performance of a forecast using some summary statistic. We will also take a deeper drive into forecasting strategies and the concept of a fan chart.

5.1 Sources of Forecasting Uncertainty

In any forecast there is some form of uncertainty. The uncertainty springs up because economic variables are inherently *random*. As such, the value of the stochastic disturbance u_t affects the forecast. Moreover, unknown parameters in the conditional mean estimates may need to be estimated. Explanatory variables may also need to be forecasted and could be random quantities themselves. Lastly, the working functional form may be mis-specified itself. We will get into each in greater detail.

5.1.1 Randomness of an Economic Variable and the Data Generating Process

Economic variables are *random processes* and therefore have a probability distribution. Recall that at the start of the lecture notes, we said that the stochastic process is the set of probability distributions of a given series accross time. In the example, we said GDP could potentially take any one of an infinite values in its probability distribution assuming that it was continuous.

We call the probability distribution the *Data Generating Process* (DGP) of that variable. If the variable has continous values, the probability of a single point forecast being equal to the eventual outcome is zero. Therefore, it is only possible to attach a probability to a range of possible outcomes encompassing the actual outcome. If for example your one step ahead forecast equals the next periods value, do not be naive that the same model will work forever. To believe that is tantamount to foolishness. Contrary to Darth Vader, your lack of faith isn't disturbing when it comes to trusting forecasting models.

5.1.2 The Value of the Disturbance and the Conditional Mean

We defined earlier that ever variable has an underlying data generating process. The DGP may be a multitude of distributions, one of which is the normal distribution $N(\mu, \sigma^2)$. However, the pdf can follow may distributions, which adds to the uncertainty, as you would imagine. For any sequence of data point y_t , we draw from this distribution. Say we want to forecast y_{t+1} , we first get its conditional mean.

$$y_{t+1} = E(y_{t+1}|\Omega_t) + u_{t+1}$$

This is the *conditional mean* where Ω_t corresponds to all known information at time t . Observant readers will note that this is the same static forecast we talked about in the naive forecasting of an *AR* or *MA*. If you recall, we said that the $E(u_{t+1}|\Omega_t) = 0$. That assumption, while I did not mention earlier, is highly restrictive. One might ask why we look at the conditional mean? The reasoning is quite simple. For one, the density function of y_{t+1} is far more difficult to estimate precisely requiring a considerable amount of data in practice. Moreover, conditional means may be estimated more readily using a variety of statistical procedures. For example, it is often assumed that the conditional mean is a linear function of other explanatory variables and we use OLS to estimate the unknowns in that relationship.

If you recall, the OLS procedure is a conditional mean procedure.

$$y_t = X_t\beta_t + u_t$$

Effectively, the estimated value of y_t which is \hat{y}_t is merely the conditional mean $E(y_t|X_t)$ as we are estimating a value for y_t given things we know can explain it X_t .

If we assume that the conditional mean is known, we still need an estimate for u_{t+1} to complete the forecast. What we did in Naive forecasting before is that, assuming u_t is a white noise process, then we can conclude that $E(u_t) = 0$. Hence, our best guess is that all of its future values is zero. Even if we know the conditional mean, our conditional forecasts will necessarily differ from the outcome. Indeed, the observed forecast errors will obviously reflect the distribution of the error term, which we assume to be normal but it is unknown. The best case that we can hope for is that, on the average, the actual forecasts values are of the error term are zero. This stems from the *law of large numbers*.

If you recall that $E(u_t) = \frac{\sum u_t}{T}$. Therefore, as $T \rightarrow \infty$

$$\frac{\sum_{t=1}^T u_t}{T} \rightarrow 0, \text{ as } T \rightarrow \infty$$

This expected value goes closer and closer to zero with a probability of 1.

5.1.3 Using Explanatory Variables

So far we have assumed that we know the conditional mean and that the conditional mean is correctly specified. However, there is a possibility that the forecast variable depends on other explanatory variables, X

$$y_{t+1} = F(X_{t+1}; \Theta) + u_{t+1}$$

The explanatory variables might be jointly determined variables (i.e. estimated within the same system or model) or determined exogenously (outside the system). For example, say we want to forecast GDP the next period. To forecast GDP, one might consider future inflation, current account, deficit, personal consumption, world GDP, and a multitude of other factors. Those factors are the explanatory variables X_t which may prove useful in forecasting GDP. However, since we will be using future values of these factors, there is a chance we would need to forecast these factors too. For example, consider our main objective is the model to follow.

$$y_{t+1} = F(X_{t+1}; \Theta) + u_{t+1}$$

For us to forecast this model, the right hand side or the explanatory variables X_t will need to be estimated or forecasted using additional equations, namely

$$X_{t+1} = F(Z_{t+1}; \Psi) + \varepsilon_{t+1}$$

These explanatory variables might require other explanatory variables Z_t and those variables might need to be forecasted as well. As such, the chance of re-cursiveness and dependency in the system causes much uncertainty which leads to greater forecast errors and the possibility for wide margins of error.

5.1.4 Complications in the Parameters and the Conditional Mean

Up to this point we assumed that the parameters we estimate in the DGP are known. That is, we know what Ω , Θ and Ψ are. However this may not always be the cause (obviously). We will continue for now the assumption that we know what the conditional mean is but assume now that we don't know the values of the parameters on X_t namely Θ .

We would need to conduct some regression $y_{t+1} = F(X_{t+1}; \Theta) + u_{t+1}$ which estimates the values of $\hat{\Theta}$. These unknown parameters need to be estimated from the available data which are also random. It

follows that the estimates of $\hat{\Theta}$ are necessarily random variables and contribute to forecast errors. The reason why they contribute to the forecast errors is that the estimates of the unknown parameters will be typically different from the true values, that is $\Theta \neq \hat{\Theta}$.

To relax the last assumption, there is a chance that the conditional mean is misspecified. This occurs because the set of explanatory variables X_t may be incomplete or faulty in explaining the variation in y_t . Moreover, the actual functional form used may be wrong. Up to now, we have only been using linear functional forms but there is a chance that the true functional form would be non-linear in nature. Lastly, the underlying parameters change over time. No one model holds true forever and parameter estimates may be erroneous or estimated incorrectly in some cases.

5.2 Measures of Forecast Uncertainty

As we have highlighted in the last subsection, there are many sources of forecast uncertainty. The challenge now lies in measuring this uncertainty. In this subsection, we will introduce measures which measure forecast uncertainty in practice. What you will find is that there is no one magic bullet that beats them all, instead, we have a host of tests and indicators which we can use to measure this.

5.2.1 Common Statistical Measures for Forecast Uncertainty

In general, we have three main statistical measures by which we measure uncertainty. That is, the bias, the variance, and the mean squared error.

The *bias* is the difference between the forecasts and the correct value (on the average). Mathematically, the difference between the forecast and the correct outcome is given as the form to follow where \hat{y}_t is the forecasted value and y_t is the correct or actual outcome.

$$FE_t = \hat{y}_t - y_t$$

To compute for the bias, we merely sum up all the FEs or the forecast errors and divide by the number of forecasts.

$$BIAS = \frac{1}{f} \sum_{t=1}^f FE_t$$

The *variance* measure is typically represented by the *Standard Forecast Error* (SE). This measure measures how spread forecasts are from each other, like a typical variance would do. Ideally, we want forecasts to not be too spread out suggesting a lower margin of error and a more precise estimate. The SE is computed for in the form to follow.

$$SE = \sqrt{\frac{1}{f} \sum_{t=1}^f (FE_t - BIAS)^2}$$

The *mean squared forecast error* (MSE) is a combination of bias and variance that is commonly reported in forecast comparisons. As this is a combination of both a bias and a variance, a lower MSE precludes a generally better forecast as we would want to lower the difference between the actual and forecast as well as the variance among forecasts.

$$MSE = \frac{1}{f} \sum_{t=1}^f FE_t^2$$

Some forecasters also use the *root mean squared error* (RMSE) to standardize the MSE.

$$RMSE = \sqrt{\frac{1}{f} \sum_{t=1}^f FE_t^2}$$

Moreover, the *mean absolute error* (MAE) and the *mean absolute percentage error* (MAPE) have also been crafted for the same standardization practice.

$$MAE = \frac{1}{f} \sum_{t=1}^f |FE_t|$$

$$MAPE = \frac{1}{f} \sum_{t=1}^f \left| \frac{FE_t}{y_t} \right|$$

Of course, some statistics have a few advantages over the other. As we mentioned previously, there is no one "magic bullet" that will give us all the knowledge on the accuracy and precision of a model. Instead, it is best to present a slew of indicators and choose depending on the nature of what we know about y_t .

5.2.2 Bias, Variance, and Covariance Proportion

The Mean Squared Error can be broken down into the bias and the standard forecast error, that is

$$MSE = SE^2 + BIAS^2$$

The form suggests that the MSE is a combination of bias and standard forecast errors. Manipulating the form above yields that MSE could also be re-expressed as the form below.

$$MSE = BIAS^2 + (s_y - s_{\hat{y}})^2 + 2(1 - r)s_y s_{\hat{y}}$$

We let s_y be the standard deviation of the actual series, $s_{\hat{y}}$ be the standard deviation of the forecasted series, and r be the correlation coefficient between the actual and the forecast series. Using this decomposition, we can derive three distinct forecast quality indicators. These indicators are proportions of the MSE, which are, the bias proportion, the covariance proportion, and the variance proportion.

The *Bias Proportion* tells us how far the mean of the forecast is from the mean of the actual series. Ideally, we would want this to be small. If the mean of the forecast is close to the actual mean, then the forecast is generally accurate.

$$BP = \frac{BIAS^2}{MSE} = \frac{\left(\sum_{t=1}^f \frac{\hat{y}_t}{f} - \sum_{t=1}^f \frac{y_t}{f} \right)^2}{\sum_{t=1}^f \frac{(\hat{y}_t - y_t)^2}{f}}$$

The *Variance Proportion* is an indicator of how far the variation of the forecast is from the variation of the actual series. Ideally, the variance proportion should be small since this suggests that the model is providing a good fit for the underlying data generating process

$$VP = \frac{(s_y - s_{\hat{y}})^2}{\sum_{t=1}^f \frac{(\hat{y}_t - y_t)^2}{f}}$$

The *Covariance Proportion* is an indicator which measures the remaining unsystematic forecasting error. For a good forecast, majority of the MSE should be due to this proportion.

$$CP = \frac{2(1 - r)s_y s_{\hat{y}}}{\sum_{t=1}^f \frac{(\hat{y}_t - y_t)^2}{f}}$$

If we sum up the three proportions, this should be equal to one as we all obtained these proportions from the MSE.

5.2.3 Symmetric Costs

The mean squared error and the absolute error assume a *symmetric cost* associated with positive or negative forecast errors. That is, the indicators don't necessarily care whether the error is positive or negative, it treats these errors equally. For example, say we forecasted y_t to be 4 and another forecast forecasted y_t to be 2. You then found out that the actual forecast was $y_t = 3$. In the first forecast, the forecast error was positive 1 (i.e. $4 - 3 = 1$) while in the second forecast, the forecast error was negative 1 (i.e. $2 - 3 = -1$). To some forecasters, those errors are similar since their deviation from the actual outcome is the same in absolute terms.

However, there is a chance that the cost of forecast errors may be asymmetric. That is, forecasters do not treat negative and positive forecasts in equal regards. Consider a central bank forecaster who is forecasting inflation for an inflation targetting central bank. The forecaster will often perceive deflation as much more costly than inflation. An airline timeliness forecaster doesn't weight delayed takeoffs equally to early takeoff as the consequences of an early take off are relatively small compared to a delayed takeoff.

5.2.4 Comparative Evaluation

It is often useful to benchmark measures against those obtained using a benchmark forecast, such as a "Naive forecast". Our economic or behavioural forecasting models should at least be as good forecasters as the established benchmark. Note however that there is often a trade-off between forecasting accuracy and the number of parameters one has to estimate in an economic model. The advantage of economic models is that it enable you to assess the reasons behind any forecast error. Comparing it to a simple "Naive" benchmark is essentially comparing the performance of your model against the obtained using a no change forecasting model. That is, for example, consider the case of a one-step ahead forecast.

$$\hat{y}_{t+1} = y_t$$

Note that the no change forecasting model merely takes the previous value as the new value and serves as an intuitive baseline. Therefore, the *MAPE* of this naive no change model is just the form to follow.

$$MAPE_{Naive} = 100 \frac{1}{f} \sum_{i=1}^f \left| \frac{y_t - y_{t-1}}{y_t} \right|$$

5.2.5 Theil's U-Statistics

Theil proposed certain indicators to measure forecast accuracy and quality. In this section, we will discuss two statistics, namely, the U_1 and U_2 . The U_1 is not normally used anymore while the U_2 is still widely used.

The U_1 statistic is a measure of *forecast accuracy* and considers the disproportionate cost of large errors. The intuition behind this statistic is quite simple. The U_1 essentially normalizes the RMSE by the dispersion of the actual and forecasted series.

$$U_1 = \frac{\sqrt{\sum_{t=1}^f \frac{(\hat{y}_t - y_t)^2}{f}}}{\sqrt{\sum_{t=1}^f \frac{\hat{y}_t^2}{f} - \sqrt{\sum_{t=1}^f \frac{y_t^2}{f}}}}$$

Note that the numerator is essentially the RMSE. Notice that U_1 will always lie within the interval $0 \leq U_t \leq 1$. More generally, this is always less than one. If the U_1 is equal to zero, then the best forecast scenario is reached. That is, there is no observation error. If the U_1 statistic is smaller for one model, it generally doesn't mean that this model is better. This is why the statistic is generally not used anymore.

The U_2 statistic is more generally used and is essentially the root mean squared percentage error relative to the error of the naive forecast.

$$U_2 = \sqrt{\frac{\frac{1}{f} \sum_{t=1}^f \left(\frac{\hat{y}_t - y_t}{\bar{y}_t} \right)^2}{\frac{1}{f} \sum_{t=1}^f \left(\frac{\bar{y}_t - y_t}{\bar{y}_t} \right)^2}}$$

In the U_2 statistic, \bar{y}_t represents the last known value of y_t . In general, if the U_2 is equal to zero, then the numerator (which is the forecast error essentially of our forecast) is zero. If U_2 is equal to one, then it means that our forecast and the naive (no change) forecast are just the same. We want smaller values of U_2 as this suggests that our forecast is better than just reusing an old value. The test is flexible in a sense that we can use another model (apart from the naive or no change model) in the denominator to, in essence, order and compare alternative forecasting models. The U_2 statistic may be used to compare the MSE of the model to any other method of producing a forecast, such as another model or even the consensus forecast. If the value of $U_2 < 1$, then the forecasting technique being used is better than just adopting the naive or no change forecast. Conversely, if the $U_2 > 1$, why did we even bother with doing a forecast since the naive model gives better results.

5.3 Forecasting Strategies and Structural Breaks

In this part we will discuss three main types of forecasting strategies. These are, the fixed window estimation, the expanding window estimation, and the rolling window estimation. In general, forecasts are made in consideration using all three strategies but let us detail how each differs from the other.

5.3.1 Fixed Window Estimation

Consider that a forecaster knows all information from time period 1 until time period T . It is most common that the forecaster will use the interval $[1, T]$ as his data to be able to forecast say $T + k$ periods ahead where k is the horizon the forecaster wants to forecast. The forecast *horizon* is the number of periods ahead a forecaster wants to estimate.

The *Fixed Estimation Window* is the most common approach. Essentially, we estimate a forecasting model using $[1, T]$ and estimate k periods ahead to the desired $T + k$ time period. Under this estimation window, the coefficients of the estimates don't change. That is, assuming the use of an *AR* or *MA*, the respective ϕ 's and θ 's do no change and are used throughout the forecasting period. Under this window or strategy, the statistics are calculated based on the values of the forecast estimates $y_{t+1}, y_{t+2}, \dots, y_{t+k}$ thus average forecast errors over different horizons. There are a few key things to bear in mind. First, what if the model performs well in the short run but not in the long run? Second, What if the model performs well in the long run but not in the short run?. How do we distinguish between these models. The answer is very straightforward and pragmatic. We use the forecast indicators we discussed and see which one produces generally the lowest indicators (i.e. lowest RMSE, lowest Theil U_2 , etc.).

5.3.2 Expanding Window Estimation

The *Expanding Window Estimation* is mostly used in short term crucial forecasts. Like the fixed window estimation, the model initially uses $[1, T]$ as the forecast period. Suppose we want to forecast $T + k$ like before. In the expanding window approach, we first use $[1, T]$ to estimate $T + k$. At time $T + 1$, we re-estimate the model using the sample $[1, T + 1]$. As the estimation window expands as we progress into the future, we get newer data and re-estimate with that new information in tow. As such, forecasts are made for the same horizon using more of actual information in the process. This is exactly what we want when we calculate for *RMSE* because the statistic will only be h steps ahead forecasts.

5.3.3 Rolling Window Estimation

The *Rolling Window Estimation* is the most robust of all (supposedly). In this method, the size of the window $[1, T]$ remains fixed like with the fixed window estimation. However, when newer periods are

realized, the window is re-adjusted to accomodate the new outcomes but maintain the length of the window. Say for example the next period $T + 1$ was realized, then, the window adjusts to $[2, T + 1]$. The length of the window is preserved but the sample period changes. As such, the coefficients will likely adjust in a proportionate manner as compared to the expanding window estimation. This estimation strategy is the most adept at accounting for *structural breaks* which we will discuss shortly.

5.3.4 Structural Breaks

When estimating a model, one key assumption is that the structure of the conditional mean doesn't change. That is, the parameter's Θ are always constant for the time periods in review.

$$y_t = F(X_t; \Theta) + u_t, \quad t = 1, \dots, T$$

However, there are certain events which may alter the series so much so that the estimates of the conditional mean would be severely affected. We call those events *structural breaks*. The most common structural breaks in economics arise from financial crisis such as the 2008 GFC, massive oil price shocks, and certainly the COVID-19 pandemic that is going on. These events have a great tendency in altering the conditional mean of the series.

More formally, a structural break is an abrupt change in parameters of the conditional mean. This abrupt change may deprecate our sample in such a manner that we have to consider two or more conditional means.

$$\begin{aligned} y_t &= F(X_t; \Theta_1) + u_t, \quad t = 1, \dots, \bar{t} \\ y_t &= F(X_t; \Theta_2) + u_t, \quad t = \bar{t} + 1, \dots, T \end{aligned}$$

In this case, \bar{t} is the time of the structural break (say 2008 for the GFC or 2020 for the COVID-19 pandemic). This would prompt a key change in the parameter Θ so much so that we need to re-estimate the model on the two time spans to be able to accurately forecast the model. Apart from this, there could be a level shift in the conditional mean. It may also be the case that the variance of the residuals change, which could cause further complications. However, our measures of uncertainty of any forecast such as the confidence intervals would need to reflect this change, otherwise, the forecasting model would be deemed incorrect. Signs of a structural break include a bias towards the *AR* coefficients in that you would find that these coefficients are very close to 1 because of the serial correlation (change in variance in the residuals caused by the structural break) among residuals. Running a unit-root test might indicate a non-stationary series even if the two sub-series are stationary.

To test for stationarity, we can use the *Chow's Test of Structural Change* which is an application of the Wald's Test of linear restrictions. To do this, we estimate three models. The first model which we refer to as the *restricted model* estimates $y_t = F(X_t; \Theta) + u_t, \quad t = 1, \dots, T$. It is restricted because we do not allow Θ to change. This suggests that the $\Theta_{pre} = \Theta_{post}$. We then estimate two additional models which we call the *unrestricted models*. The first of these is $y_t = F(X_t; \Theta_1) + u_t, \quad t = 1, \dots, \bar{t}$ and the second is $y_t = F(X_t; \Theta_2) + u_t, \quad t = \bar{t} + 1, \dots, T$. Therefore, this suggests that $\Theta_{pre} \neq \Theta_{post}$ since we allow Θ to change from Θ_1 to Θ_2 . We then estimate an *F-statistic* akin to a Wald's Test of Linear Restrictions.

$$F_{Chow} = \frac{\frac{RSS_R - (RSS_{UR1} + RSS_{UR2})}{m}}{\frac{(RSS_{UR1} + RSS_{UR2})}{T - 2k}}$$

We store the residual sum of squares (RSS) of the estimated models and compute for the F-statistic. We then let m be the number of restrictions and $T - 2k$ be the degrees of freedom. We specify our null and alternative hypothesis as follows.

$$H_0 : \Theta_1 = \Theta_2$$

$$H_a : \Theta_1 \neq \Theta_2$$

If we fail to reject the null hypothesis, it suggests that there is no structural break. This is because there was no perceived change in the conditional mean's parameters before and after the specified break. Conversely, if we reject the null hypothesis, there is evidence that a structural break occurred as a significant difference in the conditional mean's parameters was registered.

6 ARMA, ARIMA, and SARIMA

In this section, we combine our two non-stationary models into one model. It is unrealistic to think that most processes are purely an AR or purely an MA. They are likely a combination of these two processes. By combining the $AR(p)$ and the $MA(q)$, we obtain the $ARMA(p, q)$. If we add a differencing process to account for non-stationary data, we get the $ARIMA(p, d, q)$. Lastly, to account for seasonality, we can also derive the $SARIMA(p, d, q)(P, D, Q)$.

6.1 Combining AR and MA

In combining the $AR(p)$ and the $MA(q)$, we derive the $ARMA(p, q)$ model. The model is still a stationary model and only stationary data should be used when forecasting using an $ARMA$. This model combines the lagged values of the forecast variable y_t and also present and past values of a white noise error term u_t .

$$y_t = \mu + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_p y_{t-p} + \theta_1 u_{t-1} + \theta_2 u_{t-2} + \dots + \theta_q u_{t-q} + u_t$$

In more compact notation, this can be written simply as

$$\phi(L)y_t = \mu + \theta(L)u_t$$

Note that

$$\phi(L) = 1 - \phi_1 L - \phi_2 L^2 - \dots - \phi_p L^p$$

$$\theta(L) = 1 - \theta_1 L - \theta_2 L^2 - \dots - \theta_q L^q$$

We assume very similar this as the AR and MA when dealing with an $ARMA$. We assume that the residuals are white noise, with a constant variance and zero autocorrelation. The difficulty lies in distinguishing an $ARMA$ process from each other. Traditionally, we used the ACF and PACF to distinguish between models and select the optimal lags. However, as alluded to earlier, the ACF and PACF of an $ARMA$ are both geometric decay.

6.2 Autoregressive Integrated Moving Average

In the last section we mentioned something known as the integrated stochastic process. In particular, we discussed that each series has a degree of integration d which corresponds to the number of times the series has to be differenced in order to be stationary. As we just discussed, a key limitation of the $ARMA$ is that it cannot account for non-stationarity. Well, we solve that problem with the $ARIMA(p, d, q)$. We incorporate the d number of differences to stationarize the series. Essentially, this model is equivalent to an $ARMA(p, q)$ using data that had already been stationarized. Mathematically, we can represent this form as the one to follow.

$$\phi(L)\Delta^d y_t = \mu + \theta(L)u_t$$

6.3 Seasonal Autoregressive Integrated Moving Average

The $ARIMA(p, d, q)$ is certainly the workhorse model but it does have its limitations. For one, it cannot account for *seasonality*. There are some economic indicators that have seasonality. For example, consider remittances from overseas workers. The monthly remittance is usually highest during the holiday season when the expenditure of consumers tend to be high as the Christmas season prompts buying. As such, the value of the series might be high in these months compared to others in a generally recurrent manner. Hence, we can modify the standard $ARIMA$ into the Seasonal- $ARIMA$ which takes into account seasonal autoregressive lags, differences, and moving average lags. Mathematically, this is represented using the form to follow.

$$\Phi(L)\nabla^D y_t + \phi(L)\Delta^d y_t = \mu + \Theta(L)u_t + \theta(L)u_t$$

The $SARIMA(p, d, q)(P, D, Q)$ incorporates the standard $ARIMA$ specification but adds the possibility of having P seasonal autoregressive lags, D seasonal differences, and Q seasonal moving average lags.

7 Univariate Forecasting Example

In this section, we will apply all that we have learned so far to forecast the next month's inflation rate. What you should have is the *comma separated values* file (.csv) of the Monthly Inflation Rate of the Philippines from January 2000 until April 2020. The file name is *InflationRateLecture.csv*. For the full series, you can download it from the Bangko Sentral ng Pilipinas Key Statistical indicators page. You should also have a copy of R and RStudio installed on your Mac or PC. If this is your first time using R, fear not, just merely follow the codes below and you should be fine.

7.1 Preliminary Steps

To get things done, we would need to install packages and load them in R. After which, we load the dataset and transform variables into a workable series.

7.1.1 Installing and Loading Packages

The first step is to install the necessary packages. For this example, we would need to install the required packages. These are the following: tidyverse for general data manipulation and urca, forecast, TSstudio and tseries for the necessary forecasting commands. Feel free to explore the various functions you can get out of these.

Generate a new RScript so that you can keep track of things. The first order of business is installing the required packages. We can do that in the tools tab and finding the packages manually, or you can just run the commands that follow. Note that this may take some time to install depending on your internet connection.

```
install.packages("tidyverse")
install.packages("urca")
install.packages("forecast")
install.packages("tseries")
install.packages("TSstudio")
```

After installation, you must now tell R that you are about to use these packages. We do this by invoking the library() function. Consider this like taking a book from a library you want to borrow and use for a specific study session.

```
library(tidyverse)
library(urca)
library(forecast)
library(tseries)
library(TSstudio)
```

After you run all of these commands, you should be ready to run all necessary commands afterward. If in case you close your R session in the middle of forecasting, merely use the library commands once again to reload the packages. No need to reinstall (unless an update to a package is available).

7.1.2 Loading the Dataset

A simple way to load the dataset is using the `file.choose()` function which opens up a dialogue box similar to what we are accustomed to when opening files. It is important that we store the dataset in an object so we can refer to it later. First, we load the dataset and store it in an object. For the purposes of this tutorial, I'll name the object as "inflation" but you can name it anything you want.

```
inflation <- read_csv(file.choose())
```

The `read_csv` is part of the *readr* package in R inside *tidyverse*. This is used to read a csv file and to be able to load it in R. You should notice that a dialogue box should appear prompting you to choose the file. Be sure to pick *InflationRateLecture.csv*.

```
head(inflation)
nrow(inflation)
```

Running the `head(inflation)` command lists the first 6 rows of the dataset. You should see that the inflation rate in January 2000 is 5.5, February 2000 is 5.6, and so on. The `nrow(inflation)` gives the total number of rows in the dataset. This suggests that there are 244 time periods (months) in the dataset. You can also view the dataset by looking at the Environment Tab and clicking the grid beside inflation.

7.1.3 Declaring a Time Series Object

As it stands, the dataset we loaded is a *data frame*. To be able to forecast, we need a series. If you look at the dataset, we are only interested in forecasting the "Rate" variable that is in there. As such, we need to transform that "Rate" variable into a series.

```
inf <- ts(inflation$Rate, start = c(2000,1,5), frequency = 12)
```

We are creating a new object called "inf" which is the series we will use. The `ts` command creates that object. The first argument in the command is the variable. The `$` tells R that we want to get the Rate variable from the dataset inflation. Afterwards, we specify the start date which is January 5, 2000. R can automatically detect the end after this. Lastly, we specify the frequency of the series. Since we are dealing with monthly data, the frequency would be 12. If it were quarterly, frequency is 4, weekly would be 52, and bi-annual would be 2.

7.1.4 Graphing the Time Series

It is important to visualize our series. We can use the `autoplot` command to do this.

```
autoplot(inf) + ggtitle("Inflation Rate (Philippines), January 2000 to April 2020") + labs(x = "Time", y = "Inflation Rate")
```

The command plots the series "inf" with the corresponding title "Inflation Rate (Philippines), January 2000 to April 2020" where the x-axis is labelled Time and the y-axis is labelled Inflation Rate. We use the

ggtitle and the labs option for those specifications. If you did it correctly, you should have a graph that looks something like the figure to follow.

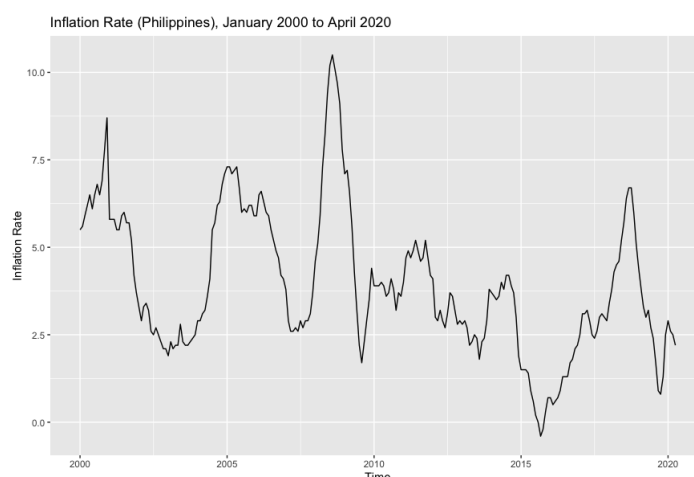


Figure 11: R Autoplot of Inflation Rate

We can clearly see the inflationary spike during the Global Financial Crisis, the Taper Tantrum, and the recent TRAIN 1 package that was passed. You can also use the `ts_plot` command to graph the same series using `plot.ly` in the `TSstudio` package.

7.1.5 Some Summary Information

We can tell R to bring up a couple of key basic statistical measures on our time series using the `summary()` command.

```
summary(inf)
```

From this, we determined that the mean inflation rate from the time span is around 3.995 percent, the highest value was 10.5 percent and the lowest value was a deflation of 0.4 percent.

7.2 Forecasting Building Blocks

We now move on to generating the ACF and PACF graph and determining whether the series is non-stationary. We use a couple of tests to detect this as outlined in the last module.

7.2.1 Generating the ACF and the PACF

Let us first generate the ACF and the PACF of the inflation rate. Again, the ACF and the ACF can tell us a lot about the properties of the series. Usually, it gives some indication as to the underlying process of the series, whether it is an *AR*, an *MA* or an *ARMA*. Furthermore, we will see what the ACF and PACF of the differenced value of inflation should we deem the series non-stationary.

```
ggAcf(inf) + ggtitle("ACF of Inflation")
ggPacf(inf) + ggtitle("PACF of Inflation")
```

The `ggAcf` and the `ggPacf` commands generate an ACF and PACF graph for our series "inf". The `ggtitle` option merely adds a title to the graphs generated.

The ACF in the figure is geometrically decaying while the PACF registers an immediate cutoff. If we recall our last lecture on the ACF and PACF, this is consistent with an *AR* process. While it is not definitive, this gives us a clue of what the underlying process might be. In the ACF, notice that the first until eleventh lag are statistically significant (beyond the blue confidence band). We can also see that some partial autocorrelations are significant such as the first, second, and fourteenth.

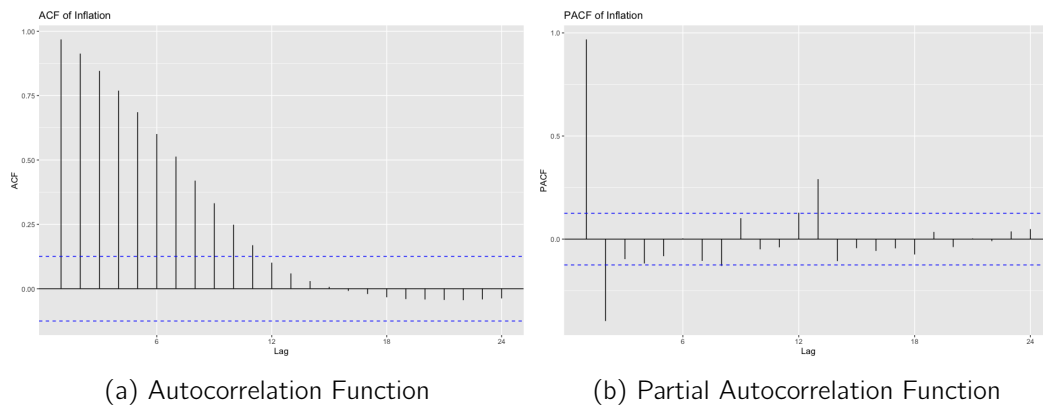


Figure 12: Showing the ACF and PACF in R

```
dinf <- diff(inf)
ggAcf(dinf) + ggtitle("ACF of Inflation (Differenced)")
ggPacf(dinf) + ggtitle("PACF of Inflation (Differenced)")
```

The first command utilizes the `diff()` function which differences the series and creates an object "dinf" which is the differenced value of inflation. By default, it differences once but we can set it to difference twice or whatever number of times specified.

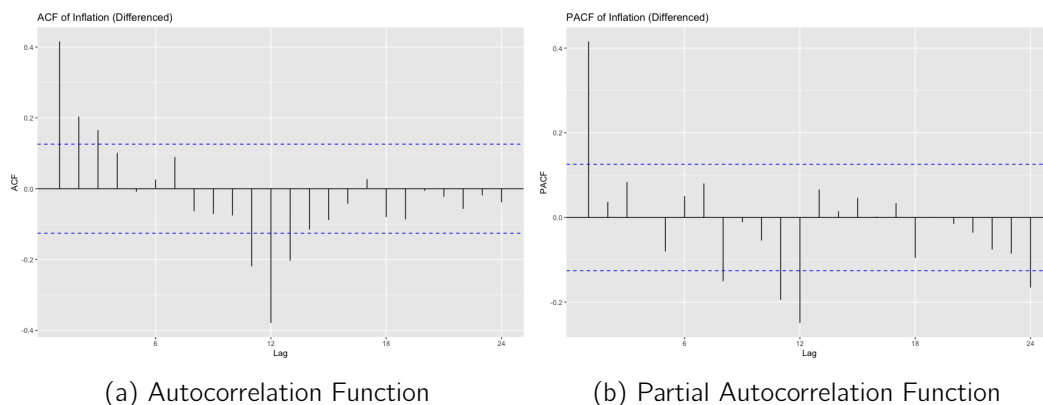


Figure 13: Showing the ACF and PACF of the Differenced in R

Similar to the levels, we can see that the ACF is geometrically decaying while an immediate cutoff is seen in the PACF. This is more evidence in support of an AR process. However, notice that not many lags are significant. This is some signal that the series has been "stationarize" so to speak which forces it to display properties of covariance stationarity. We will explore more into this as we go along.

Comparing the level values and the differenced values of inflation yields some interesting results. First, there is evidence of non-stationarity using a simple graphical look. However, we can't be too sure until we conduct formal statistical testing. Secondly, notice that there are still wild spikes (outliers) in the differenced series. These may be indicators for structural breaks which we would need to account for should they be present.

7.2.2 Decomposition of the Time Series

It is also important to see a proper decomposition of the time series we have. These components are the trend, seasonality, and random components in the series. Fortunately, there are commands in R that graph this automatically for us.

```
ts_decompose(inf, type = "additive", showline = TRUE)
```

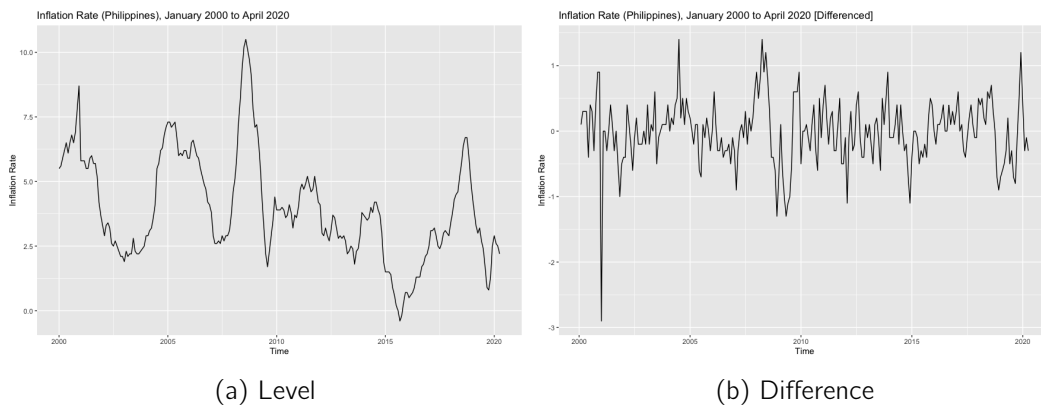


Figure 14: Showing Differenced and Level values of Inflation

The `ts_decompose` command in the library `TSstudio` gives us a good visualization of the components of the series. This command will segment the series into the observed, trend, seasonal, and random components of the series. Notice that there is some observed seasonality present, likely every August or September. The reason of which I will leave for you to figure out. The `decompose` command is in `TSstudio` is far more interactive allowing you to hover your cursor around the graph (generated by `plot.ly`, a fun graphics library) and really dig into the series. We will be using this more as we go along.

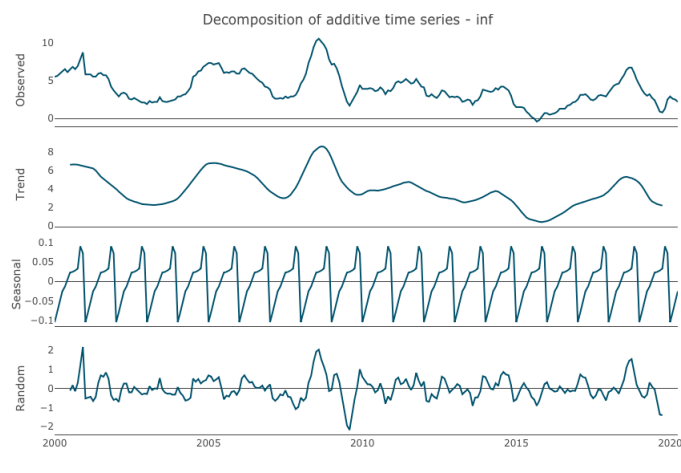


Figure 15: Decomposition of the Inflation Rate

7.3 Tests for Non-stationarity

We will use three tests which we discussed in the last section. These are, the Augmented Dickey Fuller, the Phillips Perron, and the KPSS test. Bear in mind that the ADF and PP tests are unit root tests. As such, their null hypothesis is non-stationarity while their alternative is stationary. Conversely, the KPSS test is a stationarity test in which its null hypothesis is stationarity while its alternative hypothesis is non-stationarity.

7.3.1 Augmented Dickey Fuller Test

The command to do the ADF test is `adf.test()` in R. We will use this test on the series using both the alternative hypothesis of stationarity. We will run this on both the raw and differenced series.

```
adf.test(inf)
adf.test(inf, k = 1)
adf.test(inf, k = 2)
adf.test(dinf)
```

Notice that when we ran the first line, there was a prompt suggesting that the series was stationary. The same is true for when we tested the fourth line. While inflation's differenced variable is certainly stationary, the levels value may be very deceiving. As we had said earlier, the limitation of the ADF test is that you have to specify the lag order. These tests (the first and third) were conducted in the sixth lag. If we use a more realistic lag say 1 or 2, the series was determined to be non-stationary.

7.3.2 Phillips Perron Test

For the Phillips Perron test, we use the `pp.test()` command. Unlike the ADF test, we no longer need to specify the lag order, and is more general.

```
pp.test(inf)
pp.test(dinf)
```

The results of the PP test are fairly conclusive. We see that at levels, the series is non-stationary as we fail to reject the null hypothesis. Once it is differenced, the series is stationary. As such, when we start building a forecasting model, we would need to difference inflation in order to stationarize it.

7.3.3 Kwiatkowski-Phillips-Schmidt-Shin Test

Lastly, we turn to the KPSS Test. Again, bear in mind that the null hypothesis of this test is stationarity and the alternative is non-stationarity. The command for the KPSS test is `kpss.test()`. We can specify an option called `null` to be "trend" or "level" or both. For this case, let's just leave it at the default which considers both.

```
kpss.test(inf)
kpss.test(dinf)
```

As expected, we find that the level series is not stationary. Curiously, the test also finds the differenced series as non-stationary. Still, we can refer as well to the results of the ADF and the PP tests as backup for our claim that the differenced series is stationary.

7.4 Forecasting Proper

Let us now move on to the forecasting proper. First, let us forecast in-sample before we do an out of sample forecast. As we mentioned in the first section, the reason for forecasting in-sample is to see the quality of the generated model and for us to compare different forecasting models and their estimates against actual realized values. We will use the forecast quality indicators discussed in the last chapter. Finally, towards the end, we will forecast inflation for the next year using the best forecasting model we determined.

7.4.1 Splitting the Dataset into Training and Testing Sets

Firstly, let's divide the dataset into two samples, a training sample and a testing sample. The training sample is the available data we have to forecast. The testing sample is data that is already realized (i.e. we already know the series values) but we holdout for the purposes of testing the quality of forecasting models. While before, this step used to be a pain, with R, it's a breeze. We use the `ts_split` command to do this.


```
split_inf <- ts_split(inf, sample.out = 12)

training <- split_inf$train
testing <- split_inf$test

length(training)
length(testing)
```

We create an object called `split_inf` using the `ts_split` function which is essentially `inf` but partitioned into two. The object `inf` is divided into two parts where the testing set is given by the command `sample.out`. We set the `sample.out` equal to 12 which translates to 12 months or a year. Hence, we want our testing (validation) period being equal to a year. After this, we create two objects (`training` and `testing`) which creates two separate series. Using the `length` command, we can determine how many periods there are in each of the generated series. You will notice that the training dataset has 232 periods while the testing dataset has 12 periods (exactly the same as the number specified by the `sample.out` option).

7.4.2 Diagnosing the Training Set

It is important to have at least some idea of what model to implement. To do this, we need to try and diagnose the training dataset we have to see if we can come up with suggestions as to which model may fit the series. To do this, we implore the use of the `arima_diag()` command.

```
arima_diag(training)
```

Running this yields us the figure to follow. Since this was created using `plot.ly`, it is quite interactive. We can see that it shows us the values of the training dataset as well as the associated ACF and PACF. It also shows us a graph of the differenced training series. What we can see is that the ACF is geometrically declining and an immediate cutoff is seen in the PACF. This is similar to the pattern we expect from an AR model. To determine a guess of the number of lags to use, notice that in the PACF, the first two lags are still significant. After the second lag, majority of the remaining lags are insignificant. For now, let us peg our guess at two autoregressive lags. Notice also that in the PACF and ACF, there are red datapoints. These red data points indicate the potential for a seasonal lag.

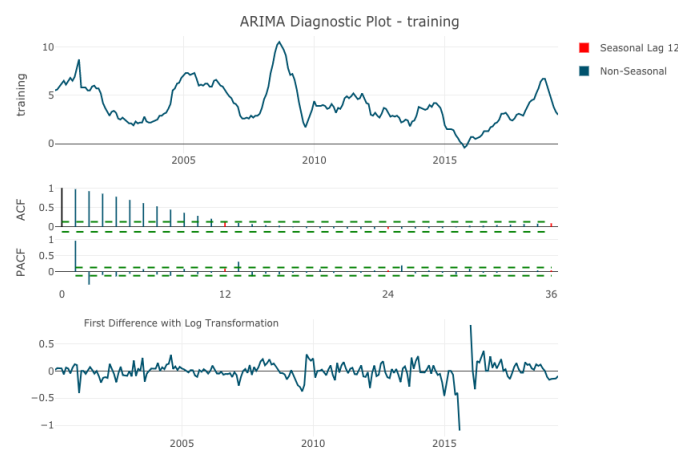


Figure 16: Diagnosing the Training Series

7.4.3 Building the Forecast Model

Let us try to forecast three models with three varying specifications. The three models we will try to use are

- An ARIMA(2,1,1)
- A SARIMA(2,1,1)(1,0,0)
- The model determined most fit using the `auto.arima()` function

For the first model, we will try to use 2 autoregressive lags since this is what the PACF is indicating. We will difference the series once because we know that at levels, the series is non stationary as seen in the results of the ADF, PP, and KPSS tests. We take a guess that we have 1 moving average lag. For the second model, we will use the same specification as the first except that we add a seasonal autoregressive lag. Lastly, for the third model, we use the built in `auto.arima()` function in R to take the guess work out of choosing which lag specification is best.

Let us move to Model 1. We create an object called `arima211` and use the `arima()` command to model. The first option in the `arima()` command is training which is the series we will be using. The order option specifies the order of the ARIMA, in this case, we specify an ARIMA with two autoregressive lags, 1 difference to stationarize (i.e. the series is integrated of order 1), and 1 moving average lag. The `autoplot()` command checks whether the stability conditions have been met. This is the stationarity condition laid out in section two. Since we are dealing with the inverse roots, all inverse roots must be *inside* the unit circle. Lastly, the `check_res` commands provides diagnostics on the residuals, which we want to be white noise. Model 2 is structured the same way except that we add a seasonal order when we invoke the `arima()` command. In that case, we add 1 seasonal autoregressive lag. Model 3 is structured the same way except we let R decide the order using the `auto.arima()` command. The option `seasonal = TRUE` just ensures that it can choose a SARIMA model if it deems it the most optimal.

```
#For Model 1
arima211 <- arima(training, order = c(2,1,1))
autoplot(arima211)
check_res(arima211)

#For Model 2
sarima2111 <- arima(training, order = c(2,1,1), seasonal = list(order = c(1,0,0)))
autoplot(sarima2111)
check_res(sarima2111)

#For Model 3
auto <- auto.arima(training, seasonal = TRUE)
auto #We obtained a SARIMA(2,1,0)(2,0,1) based on auto.arima()
autoplot(auto)
check_res(auto)
```

If you did it correctly, you should be greeted with the figures to follow. Notice that for all three models, the roots are inside the unit circle, as such, the models have passed the stationarity criterion suggesting that there are no more unit roots. Looking at the residual plots, we see that the residuals are generally white noise for all the three models. We do note however that there are some lags which are marginally outside the confidence band, but certainly model 2 and model 3 produce more white noise errors vis-a-vis model 1 which has an ACF which is quite significant at the 12th lag. This red lag is a seasonal lag which is an indication that a seasonal model is more adept.

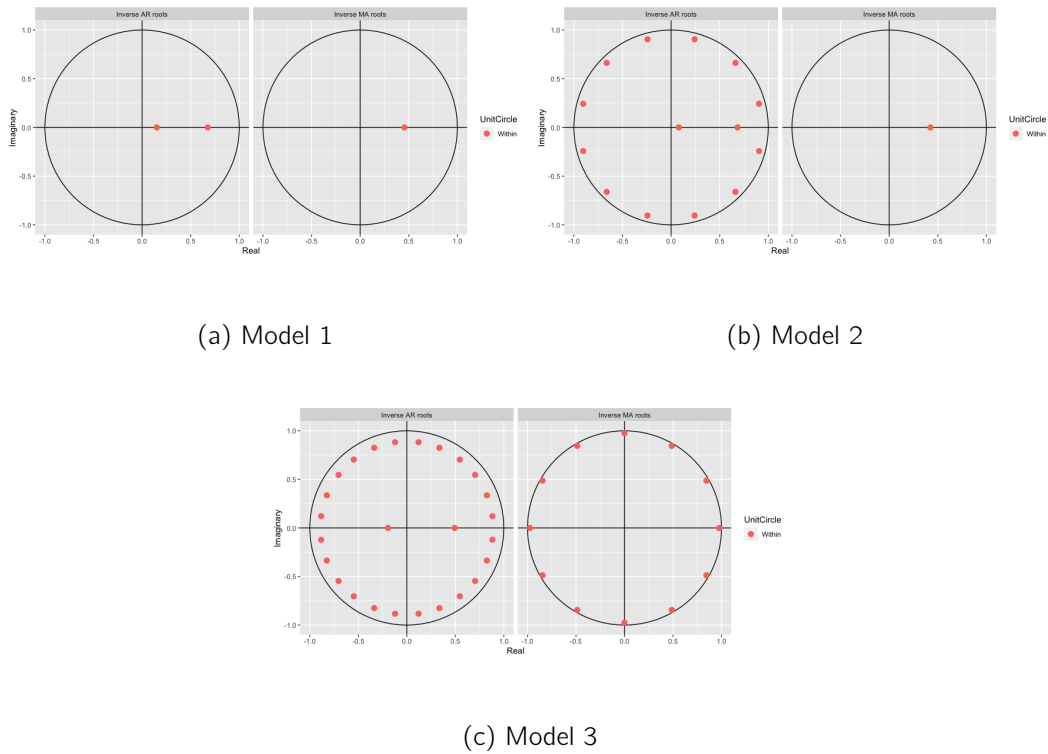


Figure 17: Inverse AR and MA Roots

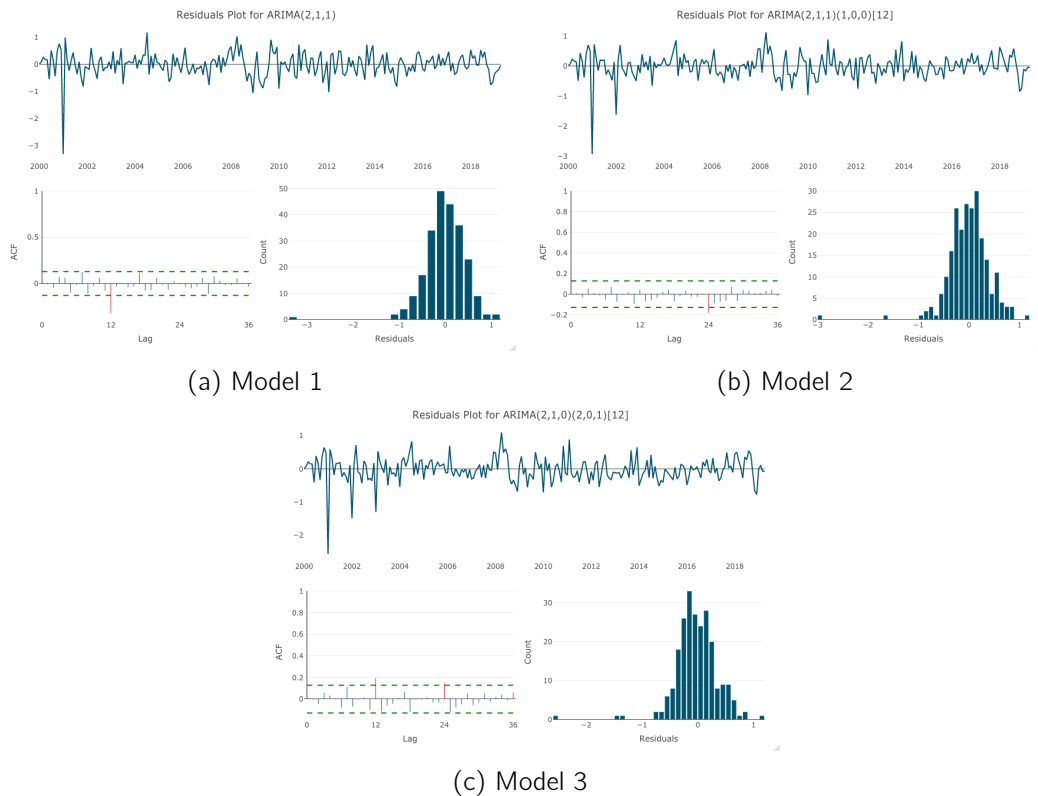


Figure 18: Residual Plots

7.4.4 Generating the Forecasts and Forecast Evaluation

Now we get to the moment of truth. Let us now generate the forecasts using each model and evaluate them against the baseline. The commands for each model are structured similarly. First, we create three

objects, namely, `fcast1`, `fcast2`, and `fcasta`, which represent the forecasts of each model. Each object is made using the `forecast()` function which has two options. The first is we key in what model it will use to forecast. The second is the number of periods to forecast. Since the test dataset is 12 periods long, we set the horizon h to 12. Next, we use the `test_forecast()` command to compare the forecast against the actual value. The command generates an interactive graph and compares each forecast against the actual value while also revealing key forecast quality indicators at every data point. Lastly, the `accuracy()` command generates the full average of the forecast indicators which we discussed in the last section. The first option in the `accuracy()` command is the forecast generated by each model being compared to the test or validation series.

We can see the graphs of the actual vs forecasting values in the figure to follow. What we notice is that it seems that Model 3 which uses the specification generated by the `auto.arima()` command seems best based on fit. Notice that when we generated these graphs, we can hover on each data point and it will give us

```
#For Model 1
fcast1 <- forecast(arima211, h = 12)
test_forecast(actual = inf, forecast.obj = fcast1, test = testing)
accuracy(fcast1,testing)

#For Model 2
fcast2 <- forecast(sarima2111, h = 12)
test_forecast(actual = inf, forecast.obj = fcast2, test = testing)
accuracy(fcast2,testing)

#For Model 3
fcasta <- forecast(auto, h = 12)
test_forecast(actual = inf, forecast.obj = fcasta, test = testing)
accuracy(fcasta,testing)
```

Further using the `accuracy()` command, we can see the forecast statistics comparison of each model. The forecast indicators are seen in the table to follow produced by the `accuracy()` command. We can see an improvement from model 1 to 2 and from model 2 to 3. This suggests that adding a season lag was vital and that the model generated by the `auto.arima()` seems to be the most optimal based on these indicators.

Table 2: Forecast Indicators

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1	Theil's U
Model 1	-0.41	0.83	0.57	-43.78	49.41	0.28	0.68	2.05
Model 2	-0.27	0.56	0.47	-24.39	31.62	0.23	0.49	1.11
Model 3	-0.24	0.36	0.3	-18.93	20.81	0.14	0.14	0.73

7.4.5 Showing the Model Gauntlet

The `TSstudio` library has a function which allows us to compare multiple forecasts at once and graph these accordingly. This is especially good if you are forecasting from a number of alternatives. Below is the code to generate the said graph. The code starts by defining an object call methods and essentially storing all models inside that object. Next, we set the forecasting horizon specifications. We set the horizon to 12 and allow for 6 rolling window partitions with a space of three months each. Once you generate the plot using the `plot_model` command, you will see a "Play" button which cycles through the six rolling window partitions. It's pretty cool. The force is strong in that code.

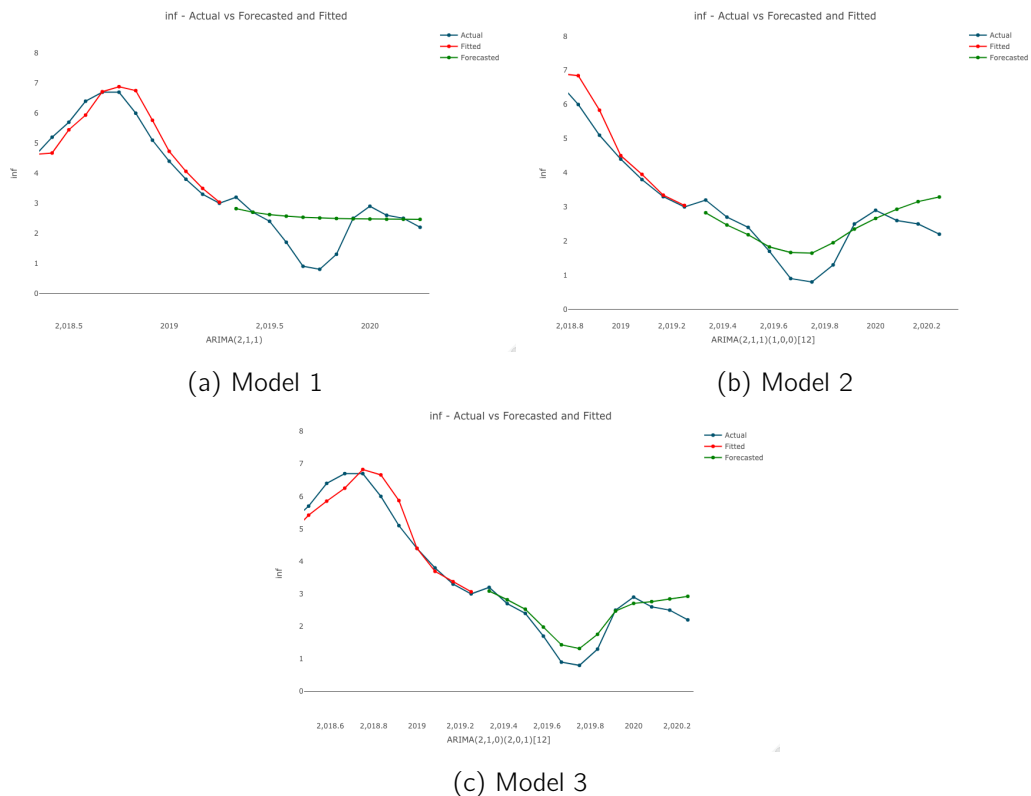


Figure 19: Forecast vs. Actual

```
# Defining the models and their arguments
methods <- list(Model1 = list(method = "arma",
method_arg = list(order = c(2,1,1)),
notes = "ARIMA(2,1,1)",
Model2 = list(method = "arma",
method_arg = list(order = c(2,1,1),
seasonal = list(order = c(1,0,0))),
notes = "SARIMA(2,1,1)(1,0,0)",
Model3 = list(method = "arma",
method_arg = list(order = c(2,1,0),
seasonal = list(order = c(2,0,1))),
notes = "SARIMA(2,1,0)(2,0,1)"))

# Training the models with backtesting
md <- train_model(input = inf,
methods = methods,
train_method = list(partitions = 6,
sample.out = 12,
space = 3),
horizon = 12,
error = "RMSE")

# Plot the models performance on the testing partitions
plot_model(model.obj = md)
```

We can see very clearly that the third model indeed resembles the actual series the most. This visualization is a great tool to have when you are comparing multiple models across different philosophies be it a pure AR or MA or some other specification.

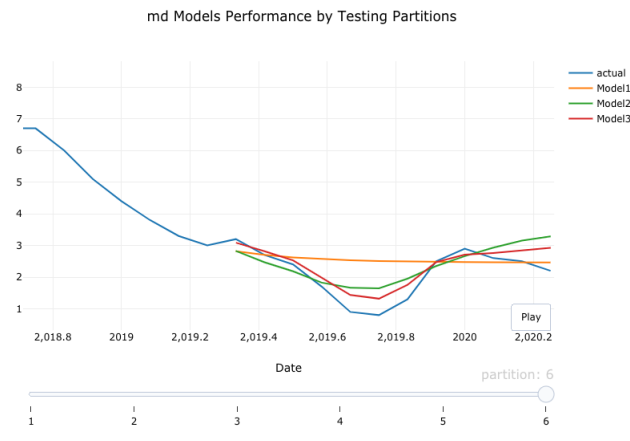


Figure 20: Comparing the Univariate Forecasting Model Specifications

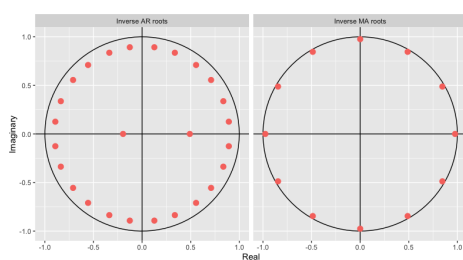
7.5 Out of Sample Forecasting

Now that we know what the most apt model is for forecasting, we shall now generate an out-of-sample forecast. In this, we don't know the actual values of the series. We can only use whatever information we have to forecast the future. This is, essentially, the forecasting you have been looking forward to.

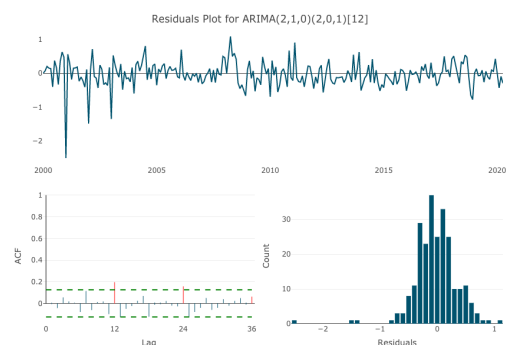
7.5.1 Generating the Optimal Fit

Similar to generating the model, we create an object wherein the model of choice is stored. Instead of using the training data, we use the whole series of inflation which is `inf`. We then test for the inverse AR and MA roots as well as the residuals.

```
finalfit <- auto.arima(inf, seasonal = TRUE)
autoplot(finalfit)
check_res(finalfit)
```



(a) Inverse AR and MA Roots



(b) Residual Diagnostics

Figure 21: Diagnostics of the Optimal Model

We find that all inverse roots are within the AR and MA unit circle. As such, we have satisfied the stability condition, there are no longer any unit roots. On the residual diagnostics, while there are still a few significant lags, it is generally white noise already.

7.5.2 Generating the Out-of-Sample Forecast

We now forecast using our chosen model. The commands below will generate the forecast. As inflation is usually re-forecasted, let us set a short horizon of 4 months ahead. Then, let us see its forecast for the

May to August 2020 inflation rates. The `summary()` command lists the point forecasts and the confidence interval for the next four months.

```
fcastf <- forecast(inf, model = finalfit, h = 4)
plot_forecast(fcastf)
summary(fcastf)
```

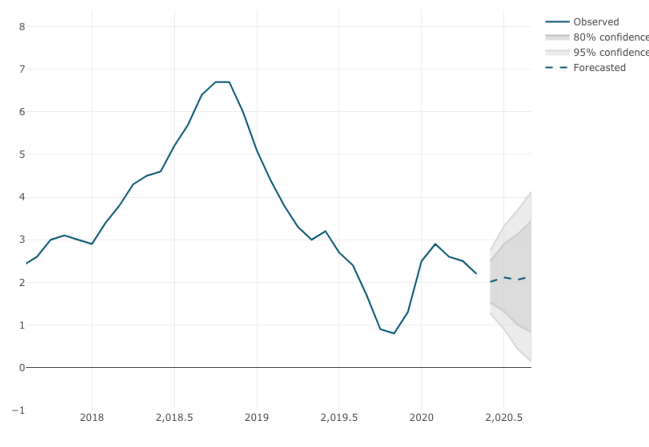


Figure 22: Forecast for May to August 2020 Inflation Rate

We can see that the forecasted values for May (2.0 percent), June (2.1 percent), July (2.1 percent), and August (2.1 percent) are fairly consistent with the BSP's inflation target and other forecast estimates. Notice too that as the horizon gets larger and larger, the confidence band expands along with it. Good job, you just forecasted your first key economic variable.

7.5.3 Some Concluding Thoughts

We have now forecasted out-of-sample and have obtained values for the next four months. We shall see if our forecasts are indeed reliable through time. Suffice to say, forecasting is a trial and error process but I hope you saw that the indicators and built in functions were able to take the guess work out of the process. Still, forecasting is also an *art* and specifying lags and conditions based on economic structure and intuition may play a pivotal role in improving forecasts. We shall see more of this as we discuss multivariate forecasting.

For now, it is recommended that you experiment with the many tools you now have learned and see if you can come up with a model which provides even better forecast quality indicators than even the model selected using the `auto.arima()` function. That is certainly possible and it just goes to show that while machines have advanced so much, there is still room for art in a place riddled full of math and science. One just needs to look for the motivation to do so.

7.6 Code Summary

```
#Forecasting Monthly Inflation
#Justin S. Eloriaga

#Installing the Required Packages

install.packages("tidyverse")
install.packages("urca")
install.packages("forecast")
install.packages("tseries")
install.packages("TSstudio")
help(TSstudio)

#Calling the Required Packages
library(tidyverse)
library(forecast)
library(tseries)
library(urca)
library(TSstudio)

#—Loading the Dataset
inflation <- read_csv(file.choose())
head(inflation)
nrow(inflation)

#—Declaring the Time Series Object
inf <- ts(inflation$Rate, start = c(2000,1,5), frequency = 12)

#Plotting the Time Series Object
autoplot(inf) + ggtitle("Inflation Rate (Philippines), January 2000 to April 2020") + labs(x =
"Time", y = "Inflation Rate")
ts_plot(inf, line.mode = "lines", title = "Inflation Rate (Philippines), January 2000 to April 2020")

#Generating some Summary Statistics

summary(inf)

#—Looking at the ACF and PACF

ggAcf(inf) + ggtitle("ACF of Inflation")
ggPacf(inf) + ggtitle("PACF of Inflation")

#Differencing the Series

dinf <- diff(inf)
```



```
#Looking at the ACF and PACF of the Differenced Series

ggAcf(dinf) + ggtitle("ACF of Inflation (Differenced)")
ggPacf(dinf) + ggtitle("PACF of Inflation (Differenced)")

#Graphing Levels and Differenced

combo <- cbind(inf, dinf)
autoplot(combo, facets = TRUE) + ggtitle("Inflation (Rate (Philippines), Level and Difference")
+ labs(y = "Rate")

#—Decomposing the Series

ts_decompose(inf, type = "additive", showline = TRUE)

#—Testing for Non-Stationarity

#Using the Augmented Dickey Fuller Test

adf.test(inf)
adf.test(inf, k = 2)
adf.test(inf, k = 1)
adf.test(dinf)

#Using the Phillips Perron Test

pp.test(inf)
pp.test(dinf)

#Using the KPSS Test

kpss.test(inf)
kpss.test(dinf)

#— In Sample Forecasting and Validation

#Partition the data into test data and training data

split_inf <- ts_split(inf, sample.out = 12)

training <- split_inf$train
testing <- split_inf$test

length(training)
length(testing)

#Using an ARIMA Diagnostic Plot on the Training Dataset

arima_diag(training)
```

```

#Trying out some Models

#For Model 1
arima211 <- arima(training, order = c(2,1,1))
autoplot(arima211)
check_res(arima211)

#For Model 2
sarima2111 <- arima(training, order = c(2,1,1), seasonal = list(order = c(1,0,0)))
autoplot(sarima2111)
check_res(sarima2111)

#For Model 3
auto <- auto.arima(training, seasonal = TRUE)
auto #We obtained a SARIMA(2,1,0)(2,0,1)
autoplot(auto)
check_res(auto)

#Forecasting Values and Diagnostics

#For Model 1
fcast1 <- forecast(arima211, h = 12)
test_forecast(actual = inf, forecast.obj = fcast1, test = testing)
accuracy(fcast1,testing)

#For Model 2
fcast2 <- forecast(sarima2111, h = 12)
test_forecast(actual = inf, forecast.obj = fcast2, test = testing)
accuracy(fcast2,testing)

#For Model 3
fcasta <- forecast(auto, h = 12)
test_forecast(actual = inf, forecast.obj = fcasta, test = testing)
accuracy(fcasta,testing)

#Graphing All Models in One

# Defining the models and their arguments
methods <- list(Model1 = list(method = "arima",
method_arg = list(order = c(2,1,1)),
notes = "ARIMA(2,1,1)"),
Model2 = list(method = "arima",
method_arg = list(order = c(2,1,1),
seasonal = list(order = c(1,0,0))),
notes = "SARIMA(2,1,1)(1,0,0)"),
Model3 = list(method = "arima",
method_arg = list(order = c(2,1,0),
seasonal = list(order = c(2,0,1))),
notes = "SARIMA(2,1,0)(2,0,1)"))

```

```
# Training the models with backtesting
md <- train_model(input = inf,
  methods = methods,
  train_method = list(partitions = 2,
    sample.out = 12,
    space = 3),
  horizon = 12,
  error = "RMSE")

# Plot the models performance on the testing partitions
plot_model(model.obj = md)

#—— Out of Sample Forecasting

#Building the Final Forecast Model
finalfit <- auto.arima(inf, seasonal = TRUE)
autoplot(finalfit)
check_res(finalfit)

#Generating the Forecast
fcastf <- forecast(inf, model = finalfit, h = 4)
plot_forecast(fcastf)
summary(fcastf)
```

End of Part I

8 Vector Autoregression

"You must feel the force around you; here, between you, me, the tree, the rock, everywhere, yes." - Yoda

Throughout majority of a student's econometrics journey, they have probably been accustomed to models functioning in this manner. The manner goes like this. Get a dependent variable. Get things you know that relate to this variable and badge them as independent. Estimate parameters which tell you how those things affect the dependent. Call it a day. That manner is pretty much what we speak of in basic econometrics. In macroeconomics, we know that certain economic variables such as inflation are dependent on the level of unemployment or the level of economic growth. We know that GDP is some function of consumption, government expenditure, investment, and net exports.

I'll draw up something you probably hear from a fortune teller. Everything around us is connected. As master Yoda said, we must feel the force around us, it connects us, binds us, and influences us. That premise is crucial in the conduct of multivariate time series. Looking at models in a single direction, that is, the independent variables affect the dependent variable, is myopic. By myopic, we mean the picture is incomplete or missing. Given that everything in the world is supposedly connected in some way, we need to free up our models to explore connections with all. That belief shall be the start of our discussion on multivariate time series which begins with the classic *Vector Autoregression* or VAR for short.

8.1 The Sim's Critique

In the early to middle of the 1900s, the prominent approach to macroeconomic modelling has been through the use of structural equations or SEM for short. This approach was spearheaded by the Cowles Commission. These structural equations are essentially equations which represent the different factors affecting key economic variables like growth or inflation. Forecasters would then estimate models using the relationship specified in a structural equations. Note that since we are talking about the whole economy, there would be dozens of structural equations to even begin to forecast. The reason for the many structural equations was that the economy is an intricate machine. The belief back then was that the economy functioned based on how economists' of that time like Keynes or Friedman had seen it to behave. Things go and influence other things in a defined manner with expected *a-priori* conditions. These conditions are expected relationships grounded on macroeconomic assumptions. Things like an increase in consumption yields an increase in inflation, and increase in unemployment decreases inflation, and so on.

The American economist and 2011 Nobel Laureate Christopher Sims, in his seminal work called "Macroeconomics and Reality" published in *Econometrica* in 1980, criticized the large scale macroeconomic models of the time. He argued that these models made strong assumptions about the dynamic nature of the relationship between macro-variables. Sims criticized the large inconsistency between the large scale modelling and the notion that economic agent take into consideration today's choices for tomorrow's expected utility. In a world with rational and forward looking agents, putting a nail down on an economic relationship is wrong. His belief is that no variable is exogenous.

Sims proposed the Vector Autoregression which is a multivariate linear time series model in which the endogenous variables in the system are functions of the lagged values of all endogenous variables. This allows for simple and flexible alternative to the traditional structural system of equations. A VAR could model macroeconomic data informatively, without imposing very strong restrictions or relationships. Essentially, it is macroeconomic modelling without much of the *a-priori* expectations getting in the way.

8.2 Uses of VARs

VARs are very useful especially in the field of macroeconomics. It has been used widely in simulating macroeconomic shocks to the real economy and has been used heavily in policy simulations and forecasting. This is the thrust and the main use of the Vector Autoregression. Firstly, it is a sophisticated *forecasting tool*. We will show how VARs can be much better than standard univariate forecasting models,

especially in determining long-run. Majority of empirical studies on forecasting suggest that the VAR has already eclipsed the traditional univariate forecasting models and theory based structural equations models.

Apart from forecasting, VARs are also useful tools for *structural analysis*. While we will discuss this in greater detail when we reach SVAR, we note that VARs are able to investigate the response to shocks. It is able to pinpoint surces of fluctuations which traditional univariate models fail at. Moreover, VARs can help distinguish between competing theoretical models.

8.3 Mathematical Foundations of the VAR

As we have mentioned previously, the VAR is a multivariate linear time series model where the endogenous variables in the system are functions of the lagged values of all endogenous variables. Said in a simple manner, the VAR is essentially a generalization of the univariate autoregressive model. Commonly, we notate a VAR as a $VAR(p)$ where p denotes the number of autoregressive lags in the system.

The simplest case of a VAR is a VAR with one autoregressive lag and with two variables in the system. Essentially, this is a bivariate $VAR(1)$. This is given in the equations to follow.

$$y_{1,t} = \alpha_{11}y_{1,t-1} + \alpha_{12}y_{2,t-1} + u_{1,t}$$

$$y_{2,t} = \alpha_{21}y_{1,t-1} + \alpha_{22}y_{2,t-1} + u_{2,t}$$

In matrix notation

$$\begin{pmatrix} y_{1,t} \\ y_{2,t} \end{pmatrix} = \begin{pmatrix} \alpha_{11} & \alpha_{12} \\ \alpha_{21} & \alpha_{22} \end{pmatrix} \begin{pmatrix} y_{1,t-1} \\ y_{2,t-1} \end{pmatrix} + \begin{pmatrix} u_{1,t} \\ u_{2,t} \end{pmatrix}$$

In compact matrix form

$$y_t = A_1 y_{t-1} + U_t$$

where

$$y_t = \begin{pmatrix} y_{1,t} \\ y_{2,t} \end{pmatrix}; A_1 = \begin{pmatrix} \alpha_{11} & \alpha_{12} \\ \alpha_{21} & \alpha_{22} \end{pmatrix}; y_{t-1} = \begin{pmatrix} y_{1,t-1} \\ y_{2,t-1} \end{pmatrix}; U_t = \begin{pmatrix} u_{1,t} \\ u_{2,t} \end{pmatrix}$$

In this case, we have two endogenous variables in the VAR system. These are $y_{1,t}$ and $y_{2,t}$. We will have to estimate one equation for each variable in the system. Notice that in each equation, say the one for $y_{1,t}$, both its own lagged value and the lagged value of the other variable in the system are present. We only include the first lag because the autoregressive order p is equal to one. The VAR is deemed a "vector" autoregression since we are essentially estimating the vector y_t which is a vector containing the endogenous variables in the system. We refer to y_t as our vector of endogenous variables, A_1 is the matrix of coefficients, y_{t-1} is the vector of lagged endogenous variables and U_t is the vector of white noise error terms.

It should be straightforward to the reader how this can scale up to any order p . For a $VAR(p)$, we merely extend the number of lagged terms. We could also opt to add an intercept in each equation which adds a vector of constants. Moreover, I hope it is clear that a p -order VAR generalizes a one-variable $AR(p)$ process for n -variables. Generalizing to a $VAR(p)$, we get the form to follow.

$$y_t = A_0 + A_1 y_{t-1} + A_2 y_{t-2} + A_3 y_{t-3} + \dots + A_p y_{t-p} + U_t$$

The equation above is what we call the *Reduced Form VAR*. The system is composed of an $(n \times 1)$ vector of constants A_0 , an $(n \times n)$ matrix of coefficients A_p , and an $(n \times 1)$ vector of white noise error terms U_t . Like with an $AR(p)$, we place certain assumptions about the error term. The following the the assumptions we place on the error term.

$$E(U_t) = 0 \quad \forall t$$

$$E(U_t U'_\tau) = \begin{cases} \Omega, & \text{if } t=\tau \\ 0, & \text{if otherwise} \end{cases}$$

The error term is still a white noise error term. First, we assume it has a mean of zero (constant) like our old assumption. This suggests that the mean is time invariant. Furthermore, we assume that the error term is not serially correlated with past error terms. The Ω is not assumed to be a diagonal matrix, that is, the error term can be contemporaneously correlated (i.e. correlated in the same time period) but it should be uncorrelated with the lagged values of the endogenous variables in the system y_t . To see this clearly, consider our two variable VAR(1) example below.

$$\underbrace{E(U_t U'_t)}_{\text{Covariance of } U_t \text{ and } U_t} = \begin{pmatrix} \sigma_{u_1}^2 & \sigma_{u_1 u_2} \\ \sigma_{u_2 u_1} & \sigma_{u_2}^2 \end{pmatrix} = \Omega$$

The terms $\sigma_{u_1}^2$ and $\sigma_{u_2}^2$ are the variances while the terms $\sigma_{u_1 u_2}$ and $\sigma_{u_2 u_1}$ are the covariances. All values inside this matrix are generally non-zero. This is because error terms are allowed to be contemporaneously correlated in the same time period. However, if we take the covariance of U_t and U_τ , that is, on two different periods, that should be a null matrix.

$$\underbrace{E(U_t U'_\tau)}_{\text{Covariance of } U_t \text{ and } U_\tau} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$$

When we say reduced form VAR, we are pertaining to a VAR with no economic restrictions and having residuals which are not orthogonal and thus cannot be interpreted as fundamental or structural shocks. In essence, it is purely *a-theoretic*. We will, in a sense, make this more rigid by introducing a structure and imposing restrictions in SVAR. That way, shocks and innovations may be interpreted to some extent in an economic context.

The estimation of a VAR is an *Equation by Equation OLS*. We essentially run OLS on each equation. What we find is that estimates are consistent as only the lagged values of the endogenous variables are on the right hand side of the equation. Moreover, estimates are also efficient in that all equations have identical regressors which minimizes the variation in each one. Doing this suggests that it is essentially equivalent to a *generalized least squares*. The assumption of no-serial correlation also holds in this regard.

8.4 Stationary VAR

The original VAR as envisioned by Sims in 1980 didn't necessarily have to be stationary or at least was designed such that stationarity wasn't a hard pre-requisite. However, as we will see, having a stationary VAR allows us to have certain properties which not adhering to may not attain.

8.4.1 Selection of Variables in a VAR

The selection of variables in a VAR is crucial as this will of course directly affect the model estimates. Theoretically, we choose in accordance with economic theory and empirical evidence. For example, consider a model for inflation targeting which most central banks use. These central banks often incorporate the output gap, inflation, and the real interest rate in the model (ala *Taylor Rule*). For small open economy models which are heavily reliant on trade, we may consider the real exchange rate.

There are also many variants of the VAR which can accomodate different conditions. First, there are VARs which may have a constant, time trends, and other additional variables. These variables may be specified as "truly" exogenous thus making the model a *VAR-X*. We will explore more of this in SVAR.

Note that in most VARs we will use, non-stationary level data is often transformed through the use of log-levels, log-differencing, or converting these variables in growth rates. The reason for converting these to stationary will be discussed soon.

In an ideal case, the model should be as parsimonious as possible, like our belief with the univariate models. The model should be estimated with the lowest possible number of suitable parameters for the economic question at hand. The reasoning for this is that VARs are *densely parametrized* in comparison with other time series models. This makes for one problem of the VAR in that it is hungry for the degrees of freedom.

8.4.2 Establishing Stationarity

Establishing whether a VAR is stationary or not has important implications. First, understanding whether a VAR is stationary has implications as to how the endogenous variables in the system respond to innovations in a gradually decaying manner, tending back to a steady state in the long run.

A p^{th} order VAR is considered to be covariance stationary if the following two conditions hold.

$$E(y_t) = E(y_{t+j}) = \mu = \begin{pmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_n \end{pmatrix}$$

The first condition is that the mean of all endogenous variables in the system are the same across time. This is similar to the constant mean assumption imposed in a simple autoregressive model.

$$E[(y_t - \mu)(y_{t+j} - \mu)] = E[(y_\tau - \mu)(y_{\tau+j} - \mu)'] = \Gamma_j$$

Note that $t \neq \tau$ which means that they are on different periods. The second condition suggests that the covariance matrix of y_t and y_{t+j} depends on the time lapsed j and not on the reference period t . Notice that the matrix Γ_j does not depend on the reference period t , rather, just the time lapsed j . This suggests that the first and second order moment conditions of y_t are finite and time-invariant.

Observant readers should have realized that the conditions are the same for an $AR(p)$. Recall that our reduced form VAR is given as

$$y_t = A_0 + A_1 y_{t-1} + A_2 y_{t-2} + A_3 y_{t-3} + \dots + A_p y_{t-p} + U_t$$

We can rewrite this in the lag operator notation form

$$(I_n - A_1 L - A_2 L^2 - A_3 L^3 - \dots - A_p L^p) y_t = A_0 + U_t$$

We can further rewrite this in the lag polynomial form

$$\underbrace{A(L) y_t}_{\text{Lag Polynomial}} = A_0 + U_t$$

8.4.3 Wold's Decomposition Representation

In order for a VAR to be stationary, the polynomial of the lag operator $A(L)$ must be *invertible*. That polynomial is invertible if all the np roots of the characteristic polynomial (in modulus) are outside the imaginary unit circle. In general, this is a generalization of the characteristic equation and the conditions of stationarity in the $AR(p)$ model. This condition pertains to the determinant of the characteristic polynomial.

$$\det(I_n - A_1 L - A_2 L^2 - A_3 L^3 - \dots - A_p L^p) \neq 0$$

If a VAR is stationary, the y_t vector can be expressed as a sum of all the past white noise shocks U_t using a $VMA(\infty)$ representation. For example, consider a $VAR(p)$ which is invertible and has an intercept μ . Then, we can represent the reduced as the form to follow.

$$y_t = \mu + A(L)^{-1}U_t$$

$$y_t = \mu + (I_n + \psi_1 L + \psi_2 L^2 + \psi_3 L^3 + \dots)U_t$$

This is the same as writing it as

$$y_t = \mu + U_t + \psi_1 U_{t-1} + \psi_2 U_{t-2} + \dots$$

Hence, this is merely an infinite sum of lagged error terms

$$y_t = \mu + \sum_{i=0}^{\infty} \psi_i U_{t-i}$$

This decomposition of the value of y_t is called *Wold's Decomposition Representation*. In this representation, ψ_i is an $(n \times n)$ matrix of coefficients of lagged values of the error term (moving average coefficients). These matrices describe the responses over time of each endogenous variable y_t to each sequence of shocks U_t . It is in these $VMA(\infty)$ representation that we obtain the *impulse response functions*. We will deal with those more in SVAR.

8.4.4 Lag Selection Criteria

Like in any time series model, determining the lag length is of great importance. In the case of a VAR, if p is too small, then the model will likely be poorly specified. If p is extremely long, there are too many degrees of freedom that will be lost. This is because every extra lag adds n^2 more coefficients to be estimated. In an ideal case, the number of lags should be sufficient for the residuals from the estimation to be white noise.

As mentioned earlier, if you use the VAR in its long run levels for as initially proposed by Sim (1980) and subsequently Sargent and not worry about some variables which are $I(1)$ (i.e. non-stationary at level), then the requirement that the residuals should be white noise is less important for you. The VAR is flexible in that regard as it may be some formulation which is purely *atheoretic* or could be theoretically rigid in some respects.

VARs provide a general description of linear dynamic system. In this regard, due to its being general, VARs are *densely parametrized* compared to the usual lengths of time series data. While it is okay if the data spans decades and has a higher frequency, we run into issues when data has gaps and is generally limited. This is what is known as the *curse of dimensionality*. In any given $VAR(p)$ model, we have essentially p matrices of a dimension of $(n \times n)$ each for A_1, A_2, \dots, A_p . One matrix to represent the vector of coefficients associated with any one endogenous variable's lag. Therefore, the total number of coefficients/parameters to be estimated is the amount below.

$$n + (n \times n \times p) = n(1 + (n \times p))$$

In here, n corresponds to the number of endogenous variables in the system and p represents the lag order.

As the VAR tries to estimate many parameters, when there is little data, the coefficient estimates are likely poorly estimated. The model may be deceiving its fit measures being relatively okay, however, the quality of out-of-sample forecasts are likely to be horrible. This problem is called *overfitting*. Overfitting might induce researchers to limit the number of variables they use and the number of lags they end up specifying which indirectly gives rise to another problem called *omitted variable bias*. This occurs when we leave out important variables or miss specifying important dynamics. These two problems will lead to poorer quality estimates and forecasts.

To address these problems, possible solutions include estimating a "core" VAR with rotating variables and adding or subtracting one by one the variables of interest. Another solution is to use Bayesian analysis which will force the model to conform to prior beliefs about the underlying processes such as unit roots and random walks. Lastly, we could also consider the multivariate transformations of the AIC, SBIC, and HQIC which we discussed before. Again, these information criteria suggest some form of tradeoff between fit and parsimony.

8.5 Forecasting with VAR

Like with the other time series models, VAR can certainly be used for forecasting. As a generalization of a simple $AR(p)$ model, you can expect similar properties as to how the forecasts of an autoregressive model would be derived. We let Ω_{t-1} be a matrix containing all information available up to time $t - 1$ and that the before realizations of U_t are known.

$$\Omega_{t-1} = (y_{t-1}, y_{t-2}, y_{t-3}, \dots, y_{t-T})$$

Therefore, if we want to forecast say y_t , then this is just essentially applying the conditional mean method we did during the $AR(p)$ forecasting.

$$E(y_t|\Omega_{t-1}) = \hat{A}_0 + \hat{A}_1 y_{t-1} + \hat{A}_2 y_{t-2} + \hat{A}_3 y_{t-3} + \dots + \hat{A}_p y_{t-p} + U_t$$

The forecast error can be decomposed into the sum of the unexpected innovation of y_t which is U_t and the coefficient estimation error ς .

$$\underbrace{y_t - E(y_t|\Omega_{t-1})}_{\text{Forecast Error}} = U_t + \varsigma(\Omega_{t-1})$$

Intuitively, this makes sense. The difference between the actual value (y_t) and the forecasted value for that $E(y_t|\Omega_{t-1})$ which is the forecast error is made up of just two things. First, the errors are due to the presence of unexpected disturbances or innovations U_t . These things are stochastic and random and prevent attaining a "perfect" point forecast. Another source of the error is merely the coefficients of the estimates of past data points. Since for the most part, the coefficients don't change and are not dynamic, they add up to the errors.

If however the estimator for these coefficients is consistent and estimates are based on many data observations, then the coefficient estimation error tends to be small, therefore, we can find that

$$y_t - E(y_t|\Omega_{t-1}) \cong U_t$$

This suggests that coefficient estimation errors essentially disappear for as long as the sample size is sufficiently large. Said another way, if the sample size increases to the population size, the coefficient estimation errors approach zero.

We can further iterate one period ahead further

$$E(y_{t+1}|\Omega_{t-1}) = \hat{A}_0 + \hat{A}_1 E(y_t|\Omega_{t-1}) + \hat{A}_2 y_{t-1} + \hat{A}_3 y_{t-2} + \dots + \hat{A}_p y_{t-p+1} + U_{t+1}$$

Generalizing, we can iterate this j periods forward

$$E(y_{t+j}|\Omega_{t-1}) = \hat{A}_0 + \hat{A}_1 E(y_{t+j-1}|\Omega_{t-1}) + \hat{A}_2 (y_{t+j-2}|\Omega_{t-1}) + \hat{A}_3 (y_{t+j-3}|\Omega_{t-1}) + \dots + \hat{A}_p (y_{t-p+j}|\Omega_{t-1}) + U_{t+j}$$

As you can tell, it also displays the same *infinite memory* property as the $AR(p)$ model. What we generated above was actually the *dynamic* version of the VAR forecast. That is, we use the forecasted values all throughout even if the actual known values have already been realized. Say for example you know the data point y_t . If you forecasted y_{t+1} using the forecasted value of y_t contingent on what you know Ω_{t-1} , then you did a dynamic forecast. Conversely, if you used the actual value of y_t in forecasting y_{t+1} , then we did a *static forecast*.

9 Structural Vector Autoregression

"I am one with the force, the force is with me" - Chirrut Îmwe

If you watched Star Wars Rogue One (and you should), the character played by the amazing Donnie Yen known as Chirrut was widely remembered by fans. In that movie, he kept on repeating this line over and over again. He said that since the force surrounds us, we are one with the force. Well, you might be wondering how this factors into our lecture? Let me ask a brief philosophical question. If we are one with the force, how can we tell or distinguish each one apart? Since the force is just a "blob" or a collection of all life, how can we isolate or identify who you are in the force if you are one with the force?

So far, we have discussed the (unrestricted) VAR and we know that it has significant advantages over typical univariate forecasting models. We know that it is a very *a-theoretic* way of going about things. However, this nature may prove to be misleading in some instances, especially when an expected policy action may imply another effect. Understanding and isolating the impact of shocks and decomposing these impacts are the thrust of the *Structural Vector Autoregression* which shall be our focus in this section.

9.1 Structural VAR Philosophy

The SVAR is a very flexible tool especially for the analysis of policy actions on the real economy. However, we know that policy can be segmented into multiple facets. For this case, we will zero in on the two main policy considerations, namely, fiscal and monetary policy. We will see in the discussions to follow how the assumptions of a standard unrestricted VAR can lead to misleading conclusions. We will also discuss how SVAR alleviates this concern by isolating impacts.

9.1.1 Monetary Policy Targeting

Suppose we want to consider the effects of a particular monetary policy action. For example, say the central bank is anticipating a rise in inflation. To curb this, the central bank may increase its policy rate. However, after the said increase, it happens that inflation still rises as anticipated. There is a chance that one could wrongly conclude that the policy rate hike led to a rise in inflation. This is an erroneous belief since it was an *endogenous* reaction to *expected* inflation.

Monetary policy reacted to the expected inflation so the sequences of events is reflecting, on the contrary, the impact of inflation expectations on monetary policy. This is not what we want to measure. We want to measure the impact of monetary policy on inflation expectations. The problem is that we can't tell these apart in a regular VAR. We can't tell apart the impact policy has on other variables. Simply put, this is not the correct way of measuring the effects of monetary policy.

9.1.2 Fiscal Policy Targeting

Suppose the government is anticipating a reduction in private demand caused by lockdown measures during a global pandemic. The government, in turn, increases public spending (which increases the deficit). However, output keeps declining for some time. Like with the monetary policy case, it is erroneous to think that public spending caused the output to decline. Again, fiscal policy (or policy in general) reacted endogenously to expected output. This is not the way of measuring the effects of public spending on the economy.

9.1.3 Isolating Impacts

We cannot measure the impact of monetary or fiscal policy when the policy variable is reacting to movements of the other variables. In order to measure the effects of policy, what we really want to do is to identify and isolate purely exogenous or purely independent movements or shocks to the variable of interest and see how the economy reacts to them. This is referred to as *impulse responses*. We will find out more

about these later in the section. What matters is that the philosophy underlying SVAR is that we want to trace out dynamic effects of shocks on the economy and we want a definitive identification process to isolate out shocks to see their effects on real economic variables.

9.2 Identifying a Structural VAR

The process of uncovering the structural model is called *identification*. In this stage, we want to identify totally exogenous shocks to the monetary policy rate and to fiscal spending. In essence, this isolates purely exogenous shocks and gets the responses of the endogenous variables after the economy is hit by these shocks. The author and proponent of the VAR Christopher Sims had this to say on identification.

Identification is the interpretation of historically observed variation in the data in a way that allows the variation to be used to predict the consequences of the action not yet undertaken.

After structural shocks are identified and responses are captured, we could make predictions on key economic variables. To be able to do this, we start by getting what we call the *structural model*.

9.2.1 General Overview of Identification and Imposing Restrictions

The identification procedure is about getting a structural model. A structural model isolates structural shocks and allows us to trace out the dynamics of the variables included in the VAR after one of these shocks hits the economy. To do that, we need to find a *matrix of contemporaneous relations* or a structural matrix which we will denote as \mathcal{A} . To be able to derive or obtain matrix \mathcal{A} , the general strategy is to restrict that matrix \mathcal{A} with economic intuition to recover the structural shocks and the structural parameters using the reduced-form VAR estimation. Therefore, the SVAR identification is essentially imposing restrictions on matrix \mathcal{A} based on economics to recover the structural shocks and parameters using the reduced form VAR.

The minimum number of restrictions required is the difference between the number of unknown and known elements. Let n be the number of variables in the VAR.

- *Unknown Elements.* The unknown elements are off diagonal elements of \mathcal{A} . Note that the diagonal elements are all equal to 1. Therefore, these off diagonal elements in \mathcal{A} amount to $(n^2 - n)$. This means that \mathcal{A} has $(n^2 - n)$ unknown elements. Furthermore, there are also n unknown variances of ε the total of which is n^2 . We will explore more into this in the section.
- *Known Elements.* The estimation allows us to get $\frac{n^2+n}{2}$ distinct elements contained in the symmetric variance-covariance matrix of the errors which is $EU_tU_t' = \sum_U$. The known elements are $\frac{n^2+n}{2}$ because there are n distinct elements from the diagonal plus $\frac{n^2-n}{2}$ elements off the diagonal. Therefore, the total number of known elements is

$$\frac{n + (n^2 - n)}{2} = \frac{n^2 + n}{2}$$

To be able to solve the system, we need to impose restrictions. Restrictions are based on economic theory and they apply to the contemporaneous correlation matrix \mathcal{A} . We would need to impose $\frac{n^2-n}{2}$ number of restrictions for a system to be solvable. For example, say we have a VAR with 2 variables, we would need to impose at least one restriction in the \mathcal{A} to be solvable.

$$\frac{2^2 - 2}{2} = \frac{4 - 2}{2} = \frac{2}{2} = 1$$

9.2.2 Linking the Structural and Reduced Form VAR

As we have said earlier, the SVAR is used to identify purely exogenous shocks. In the process of identification, we need to explore the characteristics of the *structural model*. Suppose we have a structural VAR with one lag, notated as $SVAR(1)$.

$$\mathcal{A}y_t = \beta_0 + \beta_1 y_{t-1} + \varepsilon_t$$

In the structural form, \mathcal{A} represents the matrix of contemporaneous coefficients and ε_t represents the structural shocks which are independent or exogenous.

Say that we have two variables in our system, namely, the output gap (y_t) and the real interest rate (r_t). We then constitute our equations in the SVAR as follows

$$y_t + a_{12}r_t = \beta_{10} + \beta_{11}y_{t-1} + \beta_{12}r_{t-1} + \varepsilon_{y_t}$$

$$a_{21}y_t + r_t = \beta_{20} + \beta_{21}y_{t-1} + \beta_{22}r_{t-1} + \varepsilon_{r_t}$$

In matrix form, we can represent these equations as

$$\underbrace{\begin{pmatrix} 1 & a_{12} \\ a_{21} & 1 \end{pmatrix}}_{\mathcal{A}} \begin{pmatrix} y_t \\ r_t \end{pmatrix} = \begin{pmatrix} \beta_{10} \\ \beta_{20} \end{pmatrix} + \begin{pmatrix} \beta_{11} & \beta_{12} \\ \beta_{21} & \beta_{22} \end{pmatrix} \begin{pmatrix} y_{t-1} \\ r_{t-1} \end{pmatrix} + \begin{pmatrix} \varepsilon_{y_t} \\ \varepsilon_{r_t} \end{pmatrix}$$

The matrix \mathcal{A} is crucial in identification. In this example, notice that it has 1's in the principal diagonal while the off diagonal elements represents the contemporaneous relations between both endogenous variables. The elements in the matrix \mathcal{A} represents the contemporaneous relations between the output gap and the real interest rate.

If we pre-multiply this equation by \mathcal{A}^{-1} , we can get our reduced form VAR

$$\mathcal{A}Y_t = \beta_0 + \beta_1 Y_{t-1} + \varepsilon_t$$

Pre-multiplying by \mathcal{A}^{-1}

$$\underbrace{\mathcal{A}^{-1}\mathcal{A}}_{I_n} Y_t = \underbrace{\mathcal{A}^{-1}\beta_0}_{A_0} + \underbrace{\mathcal{A}^{-1}\beta_1}_{A_1} Y_{t-1} + \underbrace{\mathcal{A}^{-1}\varepsilon_t}_{U_t}$$

Doing this allows us to derive our reduced form VAR which is solvable by OLS

$$y_t = A_0 + A_1 y_{t-1} + U_t$$

The matrix \mathcal{A} also relates the forecast errors of the reduced for VAR which is U_t to the structural shocks ε_t .

$$U_t = \mathcal{A}^{-1}\varepsilon_t$$

The forecast errors U are linear combinations of the structural shocks ε . In our two variable example for output gap, we can derive this as

$$U_{y_t} = \frac{\varepsilon_{y_t} - a_{12}\varepsilon_{r_t}}{1 - a_{12}a_{21}}$$

$$U_{r_t} = \frac{-a_{21}\varepsilon_{y_t} + \varepsilon_{r_t}}{1 - a_{12}a_{21}}$$

For example, say there was a surprise in GDP growth, for instance, and we usually don't know why it happened. Several structural shocks all combined could be affecting GDP. There could be monetary, fiscal, and productivity elements playing a role in such a surprise and we would like to distinguish those factors by getting the structural model. However, there exists a caveat with the structural model. The structural VAR cannot be estimated directly. The structural VAR is more of a theoretical construct, it is not generally *observable*. It is just the interpretation (economically) of historical data. What we have at hand is the time series (or the historical data). When we use these in a VAR, we run regressions of

each variable against its past and past values of other variables in the VAR. Thus, what we get are the coefficients A_t and the forecast errors U_t or the reduced form residuals.

Therefore, we need to start with the reduced form VAR and get the structural VAR

$$y_t = A_0 + A_1 y_{t-1} + U_t$$

$$\mathcal{A} y_t = \beta_0 + \beta_1 y_{t-1} + \varepsilon_t$$

The structural form isolates the exogenous shocks and measures the impact of these shocks on the variables included in the model. To do this, we need to get matrix \mathcal{A} . To get this matrix, we need to multiply the reduced form VAR by this matrix \mathcal{A} to get the structural model, shocks, and contemporaneous relations among variables.

Multiplying \mathcal{A} to the reduced form VAR

$$\mathcal{A} y_t = \mathcal{A} \underbrace{A_0}_{\mathcal{A}^{-1}\beta_0} + \mathcal{A} \underbrace{A_1}_{\mathcal{A}^{-1}\beta_1} y_{t-1} + \mathcal{A} \underbrace{U_t}_{\mathcal{A}^{-1}\varepsilon_t}$$

Simplifying, we get the structural VAR

$$\mathcal{A} y_t = \beta_0 + \beta_1 y_{t-1} + \varepsilon_t$$

9.2.3 Imposing Restrictions

In solving an SVAR, we start with the reduced form VAR and end up deriving the structural VAR. This involves identifying the matrix \mathcal{A} . The reduced form VAR for our output gap and real interest rate example, we can write the equations as follows for the reduced form VAR.

$$y_t = \alpha_{10} + \alpha_{11} y_{t-1} + \alpha_{12} r_{t-1} + u_{y_t}$$

$$r_t = \alpha_{20} + \alpha_{21} y_{t-1} + \alpha_{22} r_{t-1} + u_{r_t}$$

With this estimation, we obtain six coefficients (α 's) and the symmetric variance-covariance matrix of residuals. In total, for a reduced form VAR, we have 9 parameters to estimate. These are $\alpha_{10}, \alpha_{11}, \alpha_{12}, \alpha_{20}, \alpha_{21}, \alpha_{22}, \sigma_{u_y}^2, \sigma_{u_r}^2$, and σ_{u_y, u_r} . These 9 parameters are the six coefficient estimates, the two variances, and the covariance. This similar set of estimators should be familiar to you in a reduced form VAR.

To estimate the structural VAR, we need to look at how many parameters we are going to estimate. Recall the structural VAR equations we showed earlier.

$$\begin{pmatrix} 1 & a_{12} \\ a_{21} & 1 \end{pmatrix} \begin{pmatrix} y_t \\ r_t \end{pmatrix} = \begin{pmatrix} \beta_{10} \\ \beta_{20} \end{pmatrix} + \begin{pmatrix} \beta_{11} & \beta_{12} \\ \beta_{21} & \beta_{22} \end{pmatrix} \begin{pmatrix} y_{t-1} \\ r_{t-1} \end{pmatrix} + \begin{pmatrix} \varepsilon_{y_t} \\ \varepsilon_{r_t} \end{pmatrix}$$

We notice that the structural VAR has 10 parameters to estimate. These include the 8 parameters which are the β 's as well as a_{12} and a_{21} . Moreover, we have two variances to estimate for each variable. Notice that we have no covariance to estimate since the structural shocks were totally independent, which suggests that the covariance is zero.

The problem is that we have more unknowns than equations to estimate it. As a consequence, we need to *impose restrictions* to be able to solve the model. In the case of a two variable case, as mentioned earlier, we must impose at least 1 restriction on the structural parameters since all we have are 9 parameters (from the reduced form VAR).

The usual approach to go about this is by imposing restrictions on the contemporaneous relations among the endogenous variables of the structural model. The restrictions we impose are generally based on economics. This is what identification is all about. It is about imposing restrictions on matrix \mathcal{A} based on economic intuition. For example, say we set $a_{12} = 0$.

$$\begin{pmatrix} 1 & \mathbf{0} \\ a_{21} & 1 \end{pmatrix} \begin{pmatrix} y_t \\ r_t \end{pmatrix} = \begin{pmatrix} \beta_{10} \\ \beta_{20} \end{pmatrix} + \begin{pmatrix} \beta_{11} & \beta_{12} \\ \beta_{21} & \beta_{22} \end{pmatrix} \begin{pmatrix} y_{t-1} \\ r_{t-1} \end{pmatrix} + \begin{pmatrix} \varepsilon_{y_t} \\ \varepsilon_{r_t} \end{pmatrix}$$

In setting $a_{12} = 0$, we are imposing that the output gap y_t is not affected contemporaneously by a shock to the real interest rate r_t . If you see, r_t is absent in the first equation so r_t doesn't affect y_t contemporaneously. Hence, a shock on r_t only affects y_t with a lag. On the other equation, r_t is contemporaneously affected by shocks to from y_t and of course r_t itself.

$$y_t = \beta_{10} + \beta_{11}y_{t-1} + \beta_{12}r_{t-1} + \varepsilon_{y_t}$$

$$a_{21}y_t + r_t = \beta_{20} + \beta_{21}y_{t-1} + \beta_{22}r_{t-1} + \varepsilon_{r_t}$$

We must however be aware that by imposing restrictions on \mathcal{A} , you indirectly impose restrictions on \mathcal{A}^{-1} . Again, let us look at the reduced form structural VAR wherein we pre-multiplied by \mathcal{A}^{-1}

$$\mathcal{A}^{-1}\mathcal{A}Y_t = \mathcal{A}^{-1}\beta_0 + \mathcal{A}^{-1}\beta_1Y_{t-1} + \mathcal{A}^{-1}\varepsilon_t$$

If we get the inverse of \mathcal{A} , that is, \mathcal{A}^{-1}

$$\mathcal{A}^{-1} = \begin{pmatrix} 1 & 0 \\ -a_{21} & 1 \end{pmatrix}$$

Substituting this in matrix form following $\mathcal{A}^{-1}\mathcal{A}Y_t = \mathcal{A}^{-1}\beta_0 + \mathcal{A}^{-1}\beta_1Y_{t-1} + \mathcal{A}^{-1}\varepsilon_t$

$$\begin{pmatrix} y_t \\ r_t \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ -a_{21} & 1 \end{pmatrix} \begin{pmatrix} \beta_{10} \\ \beta_{20} \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ -a_{21} & 1 \end{pmatrix} \begin{pmatrix} \beta_{11} & \beta_{12} \\ \beta_{21} & \beta_{22} \end{pmatrix} \begin{pmatrix} y_{t-1} \\ r_{t-1} \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ -a_{21} & 1 \end{pmatrix} \begin{pmatrix} \varepsilon_{y_t} \\ \varepsilon_{r_t} \end{pmatrix}$$

Notice that imposing a restriction on the relation between the reduced form residuals or the forecast errors and the structural shocks. Since we set $a_{12} = 0$, the forecast errors of y_t which is U_{y_t} is equal to the structural shock ε_{y_t} . If we simplify the matrix above expressed in terms of the structural parameters, we should get the matrix form to follow.

$$\begin{pmatrix} y_t \\ r_t \end{pmatrix} = \begin{pmatrix} \beta_{10} \\ -a_{21}\beta_{10} + \beta_{20} \end{pmatrix} + \begin{pmatrix} \beta_{11} & \beta_{12} \\ -a_{21}\beta_{11} + \beta_{21} & -a_{21}\beta_{12} + \beta_{22} \end{pmatrix} \begin{pmatrix} y_{t-1} \\ r_{t-1} \end{pmatrix} + \begin{pmatrix} \varepsilon_{y_t} \\ -a_{21}\varepsilon_{y_t} + \varepsilon_{r_t} \end{pmatrix}$$

In parallel, the reduced form VAR estimated OLS has the matrix form

$$\begin{pmatrix} y_t \\ r_t \end{pmatrix} = \begin{pmatrix} \alpha_{10} \\ \alpha_{20} \end{pmatrix} + \begin{pmatrix} \alpha_{11} & \alpha_{12} \\ \alpha_{21} & \alpha_{22} \end{pmatrix} \begin{pmatrix} y_{t-1} \\ r_{t-1} \end{pmatrix} + \begin{pmatrix} u_{y_t} \\ u_{r_t} \end{pmatrix}$$

Using the coefficients A obtained with the reduced form VAR estimation above, we can build all equations that will allow us to estimate and solve for the structural parameters.

$$\beta_{10} = \alpha_{10}; \beta_{20} = \alpha_{20} + a_{21}\beta_{10}$$

$$\beta_{11} = \alpha_{11}; \beta_{12} = \alpha_{12}; \beta_{21} = \alpha_{21} + a_{21}\beta_{11}; \beta_{22} = \alpha_{22} + a_{21}\beta_{12}$$

$$\varepsilon_{y_t} = u_{y_t}; \varepsilon_{r_t} = u_{r_t} + a_{21}\varepsilon_{y_t}$$

With the restriction, the number of unknown parameters in the structural model is equal to the number of equations known from the estimated VAR. We substituted the results from the estimations for the 9 parameters estimated from the reduced form VAR to solve for the structural parameters as shown above. The structural shocks could also be recovered because we imposed that structural shocks to y_t are equal to the forecast error of y_t , which are obtained from the reduced form estimation which was caused by the restriction.

9.3 Policy Applications for SVAR

We will now proceed to the formal procedure in tracing the dynamic effects of the structural shocks on the endogenous variables. This means we are now going to see the origin of *impulse responses* which is a tractable way to isolate what will happen to economic variables should exogenous shocks permeate through the economy. We will also see *forecast error variance decompositions* to pinpoint the relationships among variables. Finally, we will establish a greater form of linear correlation in the form of the *Granger Causality*.

9.3.1 Impulse Response Functions

One main policy application of an SVAR (also VAR in general) is an impulse response function. The impulse response traces out the dynamic effects of structural shocks on the endogenous variables. Each response includes the effect of a specific shock on one of the variables in the system at an impact time t and subsequent periods after that ($t+1, t+2, \dots$). Building on the responses of the variables to structural shocks requires the model to already be identified with at least some of the structural parameters known. To do this, it requires that the SVAR be transformed into the sum of the shocks ε_t like that highlighted in Wold's decomposition representation.

$$y_t = \mu + \sum_{i=0}^{\infty} \psi_i \varepsilon_{it}$$

We will illustrate this clearer when we get to the example. Suffice to say, there are a couple of key thoughts on IRFs we should bear in mind. Firstly, if there are k variables in the system, we should expect a total of k^2 impulse responses to be generated. Say we had a bi-variate VAR between output gap and interest rate, there would be one IRF for output gap's impact on interest rate, one on interest rate's impact to output gap, one on output gap's impact to itself, and one on interest rate's impact on itself. Note that those total 4 IRFs and that these IRFs pertain to shocks in the exogenous disturbances ε_{y_t} and ε_{r_t} . Secondly, shocks can be permanent or momentary. If the path laid out in the IRF is ever increasing or decreasing even as t increases or the horizon moves further, there is a chance that we have a permanent shock. However, for the most part, we often see momentary shocks or deviations away from the steady state trend values. This is the case in most macroeconomic quantities.

9.3.2 Forecast Error Variance Decomposition

Knowledge of the forecast errors is useful to analyze the relationships among different variables. In a sequence of movements of a variable, the *variance decomposition* gives you the proportion of these movements to shocks to itself and the shocks to other variables. The *Forecast Error Variance Decomposition* (FEVD) has then been used as an argument about the origin of business cycles. The variance decompositions give the proportion of the movements in the dependent variables that are due to their own shocks vis-a-vis the shocks of other variables. In general, we expect much of the own series' shocks to explain majority of the forecast error variance of the series under a VAR system. Again, we will get to see this better when we do the example later on.

In our example earlier on output gap and the real interest rate, all of the one-period variance of y_t is due to ε_{y_t} . At longer horizons, the explanatory share of shocks due to ε_{y_t} will diminish. This is because y_t was placed first on the ordering and shocks to r do not affect y contemporaneously (the restriction we imposed). At longer horizons, the explaining share of the shock to y will diminish.

9.3.3 Granger Causality

An important application of VAR models would be causality tests which were first developed by Granger (1969) and answer the question "Do changes in one variable *cause* changes in another variable". The *Granger Causality Test* is an application of the Wald's Test of linear restrictions and is generally similar to how it is implemented in a simple OLS. Say we have two variables in the system, namely, y and x . If y Granger causes x , then it is highly probably that the lags of y should be significant in the equation of

x . If the same is true for the other direction, that is $x \rightarrow y$, then we can say that there exists *bi-directional* causality between y and x . If only one direction Granger causes, then we have *uni-directional* causality. If both directions do not Granger cause each other, then it means that there is *no causality* between y and x . Note however that the "causality" implied here is very soft as it only means that there is some correlation between the current value of one variable and the past values of another. It cannot imply that movements of one variables cause movements of another.

9.4 Recursive Ordering in SVAR

This construct is important since we want to know how to specify the order of the variables. In OLS, regardless of which variable is ordered first, the coefficients remain the same. In a VAR or SVAR system, it isn't as straightforward as that. In our example earlier, the residuals take on what we call as a *recursive ordering*. From our example earlier, we recall that the forecast error can be expressed in matrix form like the form below

$$\begin{pmatrix} U_{y_t} \\ U_{r_t} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ -a_{21} & 1 \end{pmatrix} \begin{pmatrix} \varepsilon_{y_t} \\ \varepsilon_{r_t} \end{pmatrix}$$

This suggests that the values of the forecast errors are as follows.

$$\begin{aligned} U_{y_t} &= \varepsilon_{y_t} \\ U_{r_t} &= \varepsilon_{r_t} - a_{21}\varepsilon_{y_t} \end{aligned}$$

Having a recursive ordering means that there is a sequential change. One can think of it as a timeline of events with shocks to the output gap hitting first, and then affecting the interest rate. The shocks to r don't affect y contemporaneously and only affects y with a lag (the first equation U_{y_t}). On the other hand, both shocks have a contemporaneous effect on the interest rate.

If r was a true policy interest rate, authorities or central banks observe y and react to it within the period. In other words, shocks to y affect r within the period. The question is by how much. That magnitude of how much is essentially the size of the coefficient a_{21} . The structural shocks ε will only be properly identified if there is economic support for the restriction imposed to identify the VAR. In our example, the virtue of $a_{12} = 0$ is justified based on the economic intuition mentioned above.

Consider another common example in monetary policy targeting. Say we have a three variable system with the output gap y_t , the real interest rate r_t , and the inflation rate π_t . These three variables have the corresponding forecast errors below.

$$\begin{aligned} U_{y_t} &= \varepsilon_{y_t} \\ U_{\pi_t} &= -a_{21}\varepsilon_{y_t} + \varepsilon_{\pi_t} \\ U_{r_t} &= (-a_{31} + a_{21}a_{32})\varepsilon_{y_t} - a_{32}\varepsilon_{\pi_t} + \varepsilon_{r_t} \end{aligned}$$

If movements of r are monetary policy actions, we could say that it's policy reacting to output and inflation within the period. All shocks affect r within the period. The policy rate only affects the variables y and π with a lag. This essentially adds restrictions such as the upper triangular of the matrix \mathcal{A} are zero.

It is easy to find \mathcal{A} and \mathcal{A}^{-1} when we do recursive ordering. Any invertible matrix can be broken into two lower triangular factors. We call these the *Cholesky Factors*. This is a numerical technique to easily estimate a recursive ordering. To find an inverse of \mathcal{A} , you can get the Cholesky factor of the variance-covariance matrix of the forecast errors. Bear in mind that this technique is a mathematical technique and not an economic one. You must also be aware of the economic restrictions you are placing in the upper triangular matrix (i.e. the zeroes there).

We can easily see that \mathcal{A}^{-1} is a Choleskey factor of the variance covariance matrix \sum_U . If we substitute $U_t = \mathcal{A}^{-1}\varepsilon_t$ into the variance covariance matrix, we will end up getting

$$E(U_t U_t') = \sum_U$$

$$E(\underbrace{\mathcal{A}^{-1}\varepsilon_t}_{U_t} \underbrace{\varepsilon_t \mathcal{A}^{-1}}_{U_t'}) = \sum_U$$

Assuming that $E(U_t U_t') = \sum_U = I$, we get that $(\mathcal{A}^{-1} I \mathcal{A}^{-1}) = \sum_U$ which is the same as the product of the two Choleskey factors $\sum_U = (\mathcal{A}^{-1} \mathcal{A}^{-1})$. The inverse of \mathcal{A} is a lower triangular matrix consistent with a recursive ordering. Note that assuming that $\sum_U = I$ implies that \mathcal{A} will not have 1's on the diagonal as we assumed so far. This only has to do with how we normalized the VAR. Either you have 1's on the diagonal of \mathcal{A} or you can impose that the variance of the structural shocks is 1.

If the correlations between the errors are low, the order is irrelevant. But usually, correlations are strong and the order matters. Deciding the ordering (deciding on thee restrictions) to be imposed is crucial and depends on the economic intuition.

10 Vector Autoregression Example

We will now apply the numerous concepts learned in VAR in an actual example. In particular, we will be using a framework developed by Sims (1992) using Philippine data. In this model, we will be using four variables. These are the following:

- Overnight Reverse Repurchase Rate (RRP) which is set by the Bangko Sentral ng Pilipinas. This is, by all accounts, the main policy rate that the Philippine central bank controls.
- M1 Money Supply which can be obtained from the BSP's website
- CPI Inflation Rate which is reported monthly by the Philippine Statistics Authority and measures the relative increase in prices based on a Laspeyres price index.
- Industrial Production which measures the value of all goods in the industrial sector.

We will first estimate a standard VAR which reflects a key economic response. It is believed in theory and by Sims that shocks to the nominal interest rate represent monetary policy shocks. A shock to the policy variable affects all other variables contemporaneously. The variable is affected by all the others within the period, and is order last. Lastly, the central bank only observes non-policy variables with a lag.

10.1 Preliminaries

We start again by installing the required packages and loading them using the `library()` command. For this part, we need to install the "vars" package which will have a host of commands necessary for us to run the VAR and SVAR and the diagnostic tests and applications to follow. We then see the plots of each variable and judge some initial conditions such as non-stationarity.

10.1.1 Initial Steps

As said, we need to install the "vars" package. Use the `install.packages("vars")` command to do this. After which, we will load this package together with our standard suite of packages and libraries for us to continue on the estimation.

```
install.packages("vars")
library(urca)
library(vars)
library(mFilter)
library(tseries)
library(TSstudio)
library(forecast)
library(tidyverse)
```

After installation and loading, it is time to load our dataset. We will use the file *Sample_SVAR.csv* which contains the data on all the variables from January 2003 until February 2020. The data is monthly and is publicly available, obtained from the BSP and PSA. We use the `read_csv()` command to read the dataset and the `file.choose()` command to open up a dialogue box for us to select our data. In this case, we place the dataset under an object named "mp". Name it whatever you want, if you'd like. We then use the `head()` command to see the first few rows of the dataset to inspect if it loaded correctly.

```
mp <- read_csv(file.choose())
head(mp)
```

Next, we need to declare each variable in the dataset as a time series using the `ts()` command. We use the `$` symbol to call a variable from the dataset. All variables start in the year 2003 with the same day and month of January 1. We set the frequency to 12 since we are dealing with monthly data.

```
lnIP <- ts(mp$lnIP, start = c(2003,1,1), frequency = 12)
lnM1 <- ts(mp$lnM1, start = c(2003,1,1), frequency = 12)
M1 <- ts(mp$M1, start = c(2003,1,1), frequency = 12)
CPI <- ts(mp$CPI, start = c(2003,1,1), frequency = 12)
RRP <- ts(mp$RRP, start = c(2003,1,1), frequency = 12)
```

We can also visualize our series using the `autoplot()` or `ts_plot()` command. As before, the `ts_plot()` command is a more interactive version using the `plot.ly` package as base.

```
ts_plot(lnIP)
ts_plot(lnM1)
ts_plot(M1)
ts_plot(CPI)
ts_plot(RRP)
```

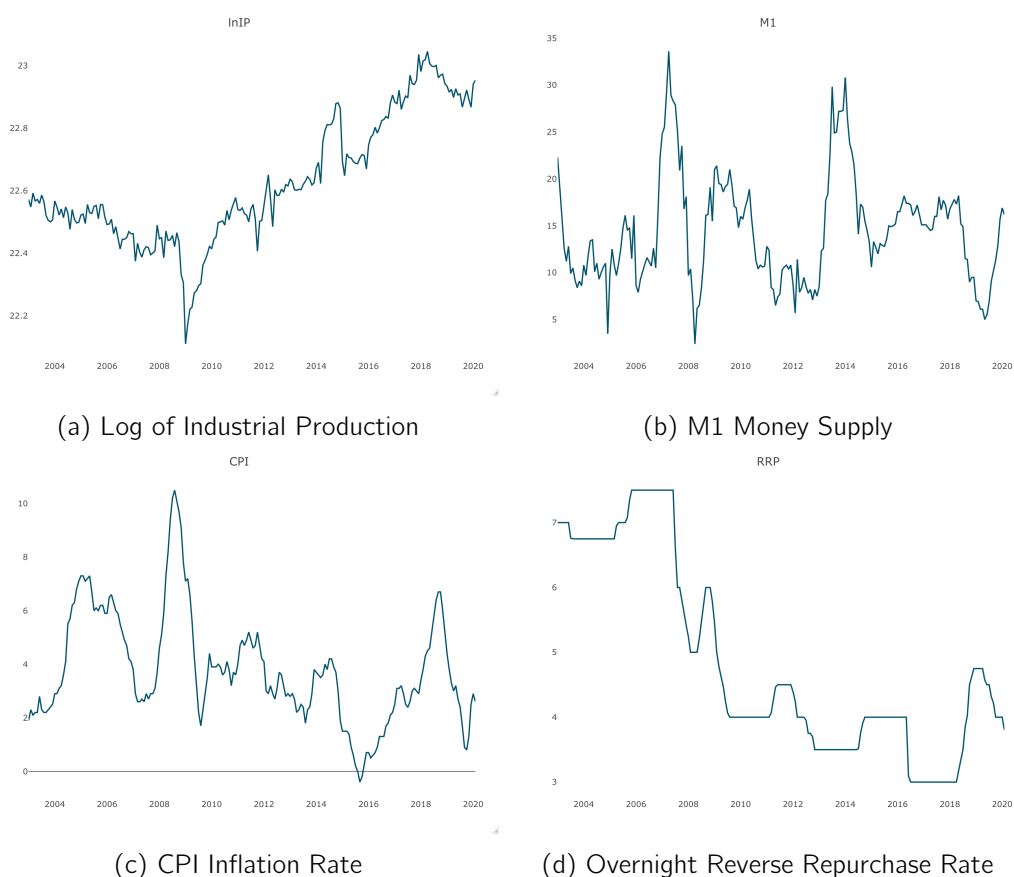


Figure 23: Time Series Plots

10.1.2 Testing for Non-Stationarity

Again, it is important to assess whether the variables under study are stationary or not. As we have said, having stationary variables is of an ideal case in our VAR even if we can run it without these. As we have been accustomed to, let us use the tests we are familiar with. For simplicity, we will use the Phillips Perron so we would not need to specify the number of lags.

```
pp.test(lnIP)
pp.test(M1)
pp.test(lnM1)
pp.test(CPI)
pp.test(RRP)
```

In the estimation above, we find that all variables with the exception of lnM1 are non-stationary variables. Remember that a rejection of the null hypothesis suggests the data is stationary. Nevertheless, we can still run a VAR estimation using these level data. However, you may opt to difference the data and see if that provides better results and forecasts.

10.2 Estimation Proper

The time has come to formally estimate our VAR. We will first need to bind our VAR variables together to create the system. After which, we will select the optimal lag order behind the VAR we will be using. We will then run an unrestricted VAR estimation and see the results. Lastly, we will run some diagnostics such as tests for autocorrelation, stability, and normality.

10.2.1 Building the VAR System

The first step is to build the VAR system. This is done through the `cbind()` command which essentially groups our time series. We will order this in the desired order that we see fit. We will store this in an object called "v1". We will then rename the variables as the list to follow using the `colnames()` command.

```
v1 <- cbind(RRP, lnM1, CPI, lnIP)
colnames(v1) <- cbind("RRP", "M1", "CPI", "lnIP")
```

10.2.2 Lag Selection

After we bind the variables and created the VAR system, we will determine some lag order which we will use. To do this, we use `VARselect()` command and use the v1 object we just created. We will use a maximum lag order of 15. The command will automatically generate the preferred lag order based on the multivariate iterations of the AIC, SBIC, HQIC and the FPE.

```
lagselect <- VARselect(v1, lag.max = 15, type = "const")
lagselect$selection
```

Running the commands suggest that the lag order to be used is 2. In the study of Sims, he used 14 lags which may have been more adept in US data as the effects reflect greater persistence.

10.2.3 Estimating the VAR Model

We will now estimate a model. We estimate the VAR using the `VAR()` command. The *p* option refers to the number of lags used. Since we determined that 2 lags was best, we set this to 2. We let it be a typical unrestricted VAR with a constant and we will specify no exogenous variables in the system. The `summary()` command lists down the results.

```
Model1 <- VAR(v1, p = 2, type = "const", season = NULL, exog = NULL)
summary(Model1)
```

We do not typically interpret the coefficients of the VAR, we typically interpret the results of the applications. You will see however that we have coefficients there for each lag and each equation in the VAR. Each equation represents an equation in the VAR system.

10.2.4 Autocorrelation

One assumption is that the residuals should, as much as possible, be non-autocorrelated. This is again on our assumption that the residuals are white noise and thus uncorrelated with the previous periods. To do this, we run the `serial.test()` command. We store our results in an object `Serial1`.

```
Serial1 <- serial.test(Model1, lags.pt = 5, type = "PT.asymptotic")
Serial1
```

In this test, we see that the residuals do not show signs of autocorrelation. However, there is a chance that if we change the maximum lag order, there could be a sign of autocorrelation. As such, it is best to experiment with multiple lag orders.

10.2.5 ARCH Test

Another aspect to consider is the presence of heteroscedasticity. In time series, there is what we call *ARCH effects* which are essentially clustered volatility areas in a time series. This is common in series' such as stock prices where massive increases or decreases could be seen when an earnings call is released. In that area or window, there could be excessive volatility thereby changing the variance of the residuals, far from our assumption of a constant variance. We have models to account for these which are the conditional volatility models which we will discuss in a later section.

```
Arch1 <- arch.test(Model1, lags.multi = 15, multivariate.only = TRUE)
Arch1
```

Again, the results of the ARCH test signify no degree of heteroscedasticity as we fail to reject the null hypothesis. Therefore, we conclude that there are no ARCH effects in this model. However, like with the autocorrelation test, it is possible to register lag effects at subsequent lag orders.

10.2.6 Normality of the Residuals

A soft pre-requisite but a desirable one is the normality of the distribution of the residuals. To test for the normality of the residuals, we use the `normality.test()` command in R which brings in the Jarque-Bera test, the Kurtosis Test, and the Skewness test.

```
Norm1 <- normality.test(Model1, multivariate.only = TRUE)
Norm1
```

Based on all the three results, it appears that the residuals of this particular model are not normally distributed.

10.2.7 Stability Test

The stability test is some test for the presence of structural breaks. We know that if we are unable to test for structural breaks and if there happened to be one, the whole estimation may be thrown off. Fortunately, we have a simple test for this which uses a plot of the sum of recursive residuals. If at any point in the graph, the sum goes out of the red critical bounds, then a structural break at that point was seen.

```
Stability1 <- stability(Model1, type = "OLS-CUSUM")
plot(Stability1)
```

Based on the results of the test, there seems to be no structural breaks evident. As such, our model passes this particular test.

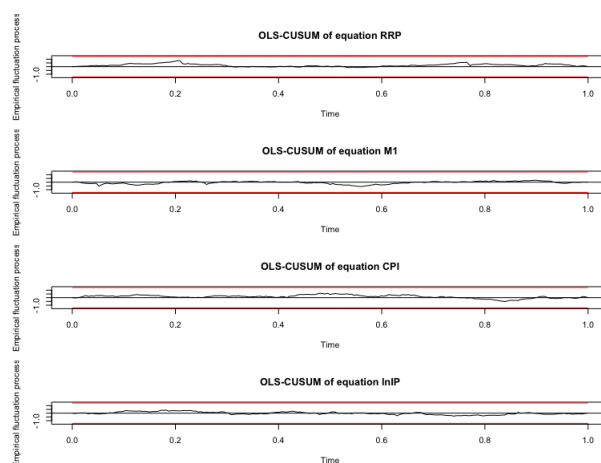


Figure 24: Stability Test in VAR

10.3 Policy Simulations

We will now move on to policy simulations in a regular VAR. We talked about three main ones, which are the Granger Causality, Forecast Error Variance Decomposition, as well as the Impulse Response Functions. While this was discussed in SVAR, this is also applicable for a regular unrestricted VAR.

10.3.1 Granger Causality

We will test for an overall granger causality testing each variable in the system against all the others. As we said, there could be a unidirectional, bidirectional, or no causality relationships between variables.

```
GrangerRRP <- causality(Model1, cause = "RRP")
GrangerRRP
```

```
GrangerM1 <- causality(Model1, cause = "M1")
GrangerM1
```

```
GrangerCPI <- causality(Model1, cause = "CPI")
GrangerCPI
```

```
GrangerInIP <- causality(Model1, cause = "InIP")
GrangerInIP
```

The command used is suitable for bivariate cases but we will use it in a system of four for now. Some of you may try reducing the VAR models to two variables to see more straightforward interpretations. To perform the Granger causality test, we use the `causality()` command. Based on the results, we conclude that CPI Granger causes the other variables but the converse is not seen. Maybe granular variable to variable rather than variable to group relationships would prove more significant.

10.3.2 Impulse Response Functions

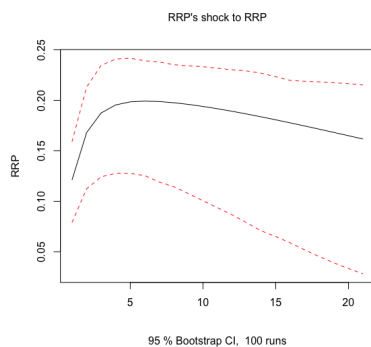
We will now turn to our impulse response functions. While we can get more impulse response functions than the ones below, we will zero in on the impact of a shock in RRP to the other variables in the system

```
RRPirf <- irf(Model1, impulse = "RRP", response = "RRP", n.ahead = 20, boot = TRUE)
plot(RRPirf, ylab = "RRP", main = "RRP's shock to RRP")
```

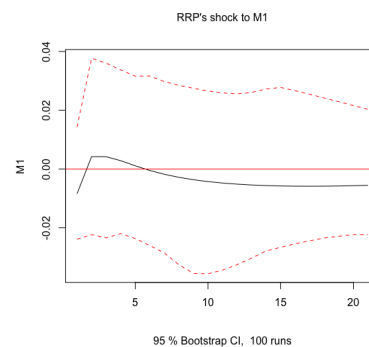
```
M1irf <- irf(Model1, impulse = "RRP", response = "M1", n.ahead = 20, boot = TRUE)
plot(M1irf, ylab = "M1", main = "RRP's shock to M1")
```

```
CPIirf <- irf(Model1, impulse = "RRP", response = "CPI", n.ahead = 20, boot = TRUE)
plot(CPIirf, ylab = "CPI", main = "RRP's shock to CPI")
```

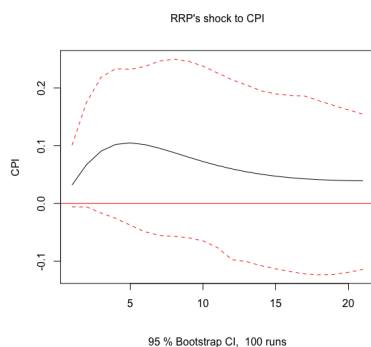
```
lnPIrf <- irf(Model1, impulse = "RRP", response = "lnIP", n.ahead = 20, boot = TRUE)
plot(lnPIrf, ylab = "lnIP", main = "RRP's shock to lnIP")
```



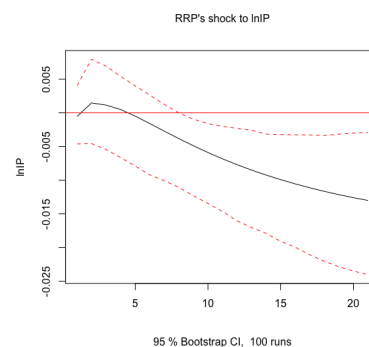
(a) Shock of RRP to RRP



(b) Shock of RRP to M1 Money Supply



(c) Shock of RRP to CPI



(d) Shock of RRP to lnIP

Figure 25: Impulse Responses in VAR

The `irf()` command generates the IRFs where we need to specify the model, what the impulse series will be, and what the response series will be. We can also specify the number of periods ahead to see how the impact or shock will progress over time. We then use the `plot()` command to graph this IRF. The results from the IRF are quite *puzzling*. Remember that such an increase in the RRP cannot be a reaction of the BSP to what is happening to the other variables since it was ordered first. As you can see, in (a), the shock to the RRP will of course increase RRP, this would then lead to a slight fall in the money supply (b) and a fall in output (d). What we also find is that prices increase in the short run but decrease moderately thereafter. According to Christopher Sims, this is a puzzling result. The results suggest that prices go up after an RRP hike. If a monetary contraction reduces aggregate demand (lnIP) thereby lowering output, it cannot be associated with inflation. He goes on to say that the VAR could potentially be miss-specified. For instance, there could be a leading indicator for inflation to which the BSP will react and which was wrongly omitted from the VAR. The BSP could know that inflationary pressures are about to arrive and counteract that by raising the interest rate.

If you recall basic macroeconomics, a contractionary monetary policy like raising the policy rate (interest rate) will lead to a fall in output. That fall in output should generally weigh inflation down and cause prices to decrease. However, what we see in this model is that prices have increased, for the most part, which is puzzling. Supposedly, exogenous movements of the policy interest rate in the previous estimation were not totally exogenous. In other words, the exogenous shocks were not properly identified.

10.3.3 Forecast Error Variance Decomposition

We now turn our attention to the forecast error variance decomposition. Again, we can trace the development of shocks in the our system to explaining the forecast error variances of all the variables in the system. To do this, we use the `fevd()` command. Like the IRF, we can also specify the number of periods ahead. For this case, let us just zero in on RRP.

```
FEVD1 <- fevd(Model1, n.ahead = 10)
FEVD1
plot(FEVD1)
```

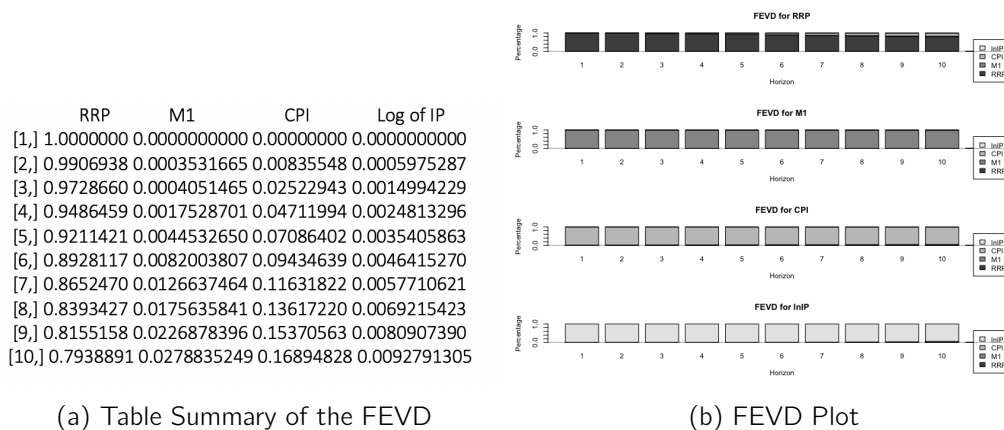


Figure 26: Forecast Error Variance Decomposition in VAR

We can clearly see in window (a) that the forecast error of the RRP (column 1) at short horizons is due to itself. This is the case because the RRP was placed first in the ordering and that no other shocks affect RRP contemporaneously. At longer horizons say 10 months, we can see that CPI now accounts for about 16 percent and that money supply accounts for about 2 percent respectively.

10.3.4 Forecasting using a VAR

As we have said, we can also forecast using VAR. To do this, we use the `predict()` command and generate something called a *fanchart* which is commonly used in identifying the confidence level of forecasts in a graphical manner. We will set the forecast horizon to 12 months ahead or a full year forecast.

```
forecast <- predict(Model1, n.ahead = 12, ci = 0.95)
fanchart(forecast, names = "RRP", main = "Fanchart for RRP", xlab = "Horizon", ylab = "RRP")
fanchart(forecast, names = "M1", main = "Fanchart for M1", xlab = "Horizon", ylab = "M1")
fanchart(forecast, names = "CPI", main = "Fanchart for CPI", xlab = "Horizon", ylab = "CPI")
fanchart(forecast, names = "lnIP", main = "Fanchart for lnIP", xlab = "Horizon", ylab = "lnIP")
forecast
```

The figure shows that RRP is expected to decrease slightly then increase, M1 is expected to decrease and so is IP. CPI is expected to decrease slightly in the first few months then rebound moderately. Take note that this is different from the IRFs simply because we are using VAR as a forecasting tool instead of a policy tool. All in all, I hope with this example you are able to see the many use cases of the VAR methodology and why many economists continue to use it for its flexibility.

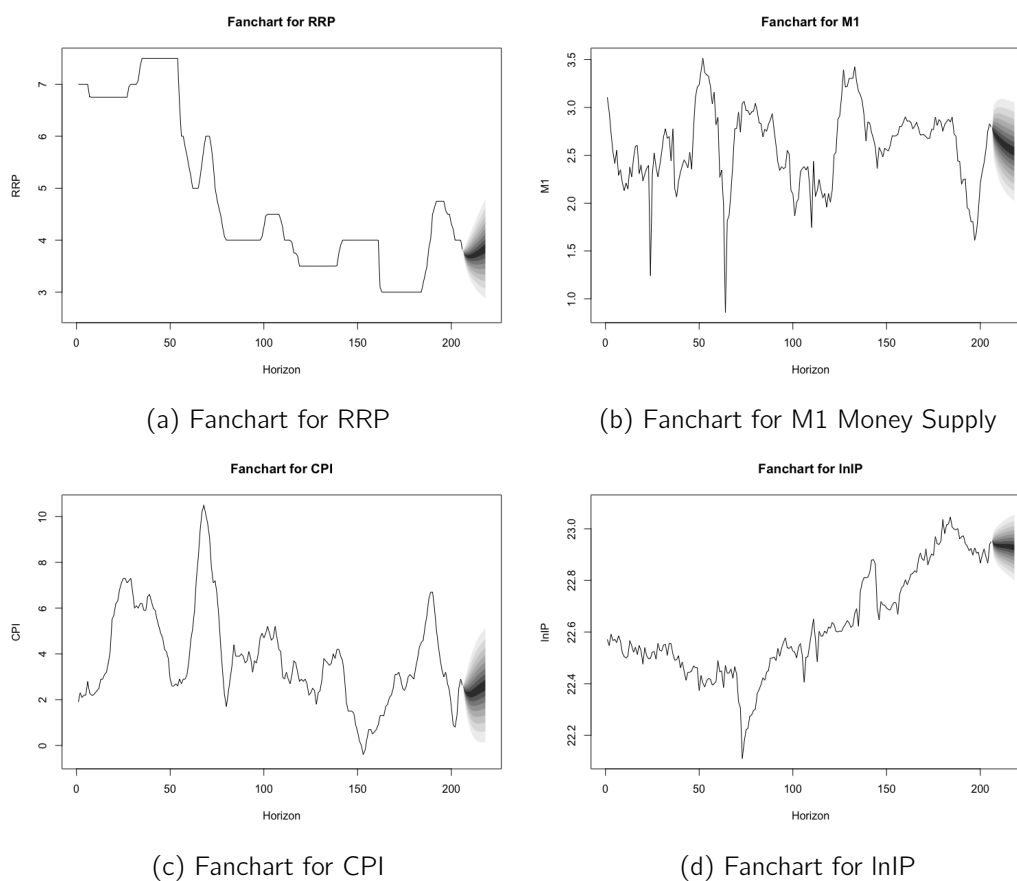


Figure 27: Fancharts in VAR

10.4 Code Summary

```
#VAR in R
#Justin S. Eloriaga

#Install the vars package

install.packages("vars")

#Load required packages for running VAR

library(urca)
library(vars)
library(mFilter)
library(tseries)
library(TSstudio)
library(forecast)
library(tidyverse)

#Load the Dataset
```

```

mp <- read_csv(file.choose())
head(mp)

#Declare and Graph our Time Series Variables

lnIP <- ts(mp$lnIP, start = c(2003,1,1), frequency = 12)
lnM1 <- ts(mp$lnM1, start = c(2003,1,1), frequency = 12)
M1 <- ts(mp$M1, start = c(2003,1,1), frequency = 12)
CPI <- ts(mp$CPI, start = c(2003,1,1), frequency = 12)
RRP <- ts(mp$RRP, start = c(2003,1,1), frequency = 12)

#Graphing the Series

ts_plot(lnIP)
ts_plot(lnM1)
ts_plot(M1)
ts_plot(CPI)
ts_plot(RRP)

#Testing for Non-stationarity

pp.test(lnIP)
pp.test(M1)
pp.test(lnM1)
pp.test(CPI)
pp.test(RRP)

#Finding the Optimal Lags

v1 <- cbind(RRP, lnM1, CPI, lnIP)
colnames(v1) <- cbind("RRP", "M1", "CPI", "lnIP")

lagselect <- VARselect(v1, lag.max = 15, type = "both")
lagselect$selection

#Building VAR
Model1 <- VAR(v1, p = 2, type = "const", season = NULL, exog = NULL)
summary(Model1)

#Diagnosing the VAR

#Serial Correlation

Serial1 <- serial.test(Model1, lags.pt = 5, type = "PT.asymptotic")
Serial1

#Heteroscedasticity

Arch1 <- arch.test(Model1, lags.multi = 15, multivariate.only = TRUE)
Arch1

```

#Normal Distribution of the Residuals

```
Norm1 <- normality.test(Model1, multivariate.only = TRUE)
Norm1
```

#Testing for Structural Breaks in the Residuals

```
Stability1 <- stability(Model1, type = "OLS-CUSUM")
plot(Stability1)
```

#Granger Causality

```
GrangerRRP<- causality(Model1, cause = "RRP")
GrangerRRP
```

```
GrangerM1<- causality(Model1, cause = "M1")
GrangerM1
```

```
GrangerCPI<- causality(Model1, cause = "CPI")
GrangerCPI
```

```
GrangerInIP<- causality(Model1, cause = "InIP")
GrangerInIP
```

#Impulse Response Functions

```
RRPirf <- irf(Model1, impulse = "RRP", response = "RRP", n.ahead = 20, boot = TRUE)
plot(RRPirf, ylab = "RRP", main = "RRP's shock to RRP")
```

```
M1irf <- irf(Model1, impulse = "RRP", response = "M1", n.ahead = 20, boot = TRUE)
plot(M1irf, ylab = "M1", main = "RRP's shock to M1")
```

```
CPIirf <- irf(Model1, impulse = "RRP", response = "CPI", n.ahead = 20, boot = TRUE)
plot(CPIirf, ylab = "CPI", main = "RRP's shock to CPI")
```

```
InPIrf <- irf(Model1, impulse = "RRP", response = "InIP", n.ahead = 20, boot = TRUE)
plot(InPIrf, ylab = "InIP", main = "RRP's shock to InIP")
```

#Variance Decomposition

```
FEVD1 <- fevd(Model1, n.ahead = 10)
FEVD1
plot(FEVD1)
```

#VAR Forecasting

```
forecast <- predict(Model1, n.ahead = 12, ci = 0.95)
fanchart(forecast, names = "RRP", main = "Fanchart for RRP", xlab = "Horizon", ylab = "RRP")
fanchart(forecast, names = "M1", main = "Fanchart for M1", xlab = "Horizon", ylab = "M1")
fanchart(forecast, names = "CPI", main = "Fanchart for CPI", xlab = "Horizon", ylab = "CPI")
fanchart(forecast, names = "InIP", main = "Fanchart for InIP", xlab = "Horizon", ylab = "InIP")
forecast
```

11 Structural Vector Autoregression Example

We will now turn to our example on the structural VAR. We will be using a roughly similar methodology to the VAR but adding some restrictions on the values and some more conditions. Fortunately, there is a library of commands in R for this very purpose. For our example, we will be dealing with three variables, namely, the output gap y_t , the inflation rate π_t , and interest rate (RRP) which is r_t . If you recall, we used this example previously in the section on SVAR but never really ran it there. We will impose the same restrictions as the conditions which we have outlined there.

11.1 Preliminaries

As always, we start by loading and installing the packages, transforming the variables to time series objects, seeing these variables graphically, and checking some general conditions.

11.1.1 Installing and Loading the Required Packages

We start by loading and installing the required packages. The packages we need to use are roughly similar to those in VAR, except we will be installing one specialized packaged called "svars". Like always, we will load the packages using the `library()` command and install using the `install.packages()` command.

```
install.packages("svars")
library(urca)
library(vars)
library(mFilter)
library(tseries)
library(TSstudio)
library(forecast)
library(tidyverse)
library(svars)
```

11.1.2 Time Series Loading

We load our dataset which is the `SVAR_Philippines.csv` file. This contains the data points in our study which run from Q1 2000 to Q1 2020. The frequency of this data is quarterly. All datapoints are available at the BSP website except for the Output Gap which was estimated by the author using a *Kalman Filtering* technique.

```
macro <- read_csv(file.choose())
head(macro)
```

After loading the dataset, we need to turn our variables into time series variables useable for our SVAR. Like always, we use the `ts()` command to do this.

```
y <- ts(macro$'Output Gap', start = c(2000,1,1), frequency = 4)
pi <- ts(macro$CPI, start = c(2000,1,1), frequency = 4)
r <- ts(macro$RRP, start = c(2000,1,1), frequency = 4)
```

11.1.3 Plotting the Series

As always, we can plot each time series using the `ts_plot()` command. It is important to visualize the series to verify the loading of the series and to get a feel for certain characteristics. You may opt to use the `ts_decompose` command to see a better decomposition.

```
ts_plot(y, title = "Output Gap", Xtitle = "Time", Ytitle = "Output Gap")
ts_plot(pi, title = "Inflation Rate", Xtitle = "Time", Ytitle = "Inflation Rate")
ts_plot(r, title = "Overnight Reverse Repurchase Rate", Xtitle = "Time", Ytitle = "RRP")
```

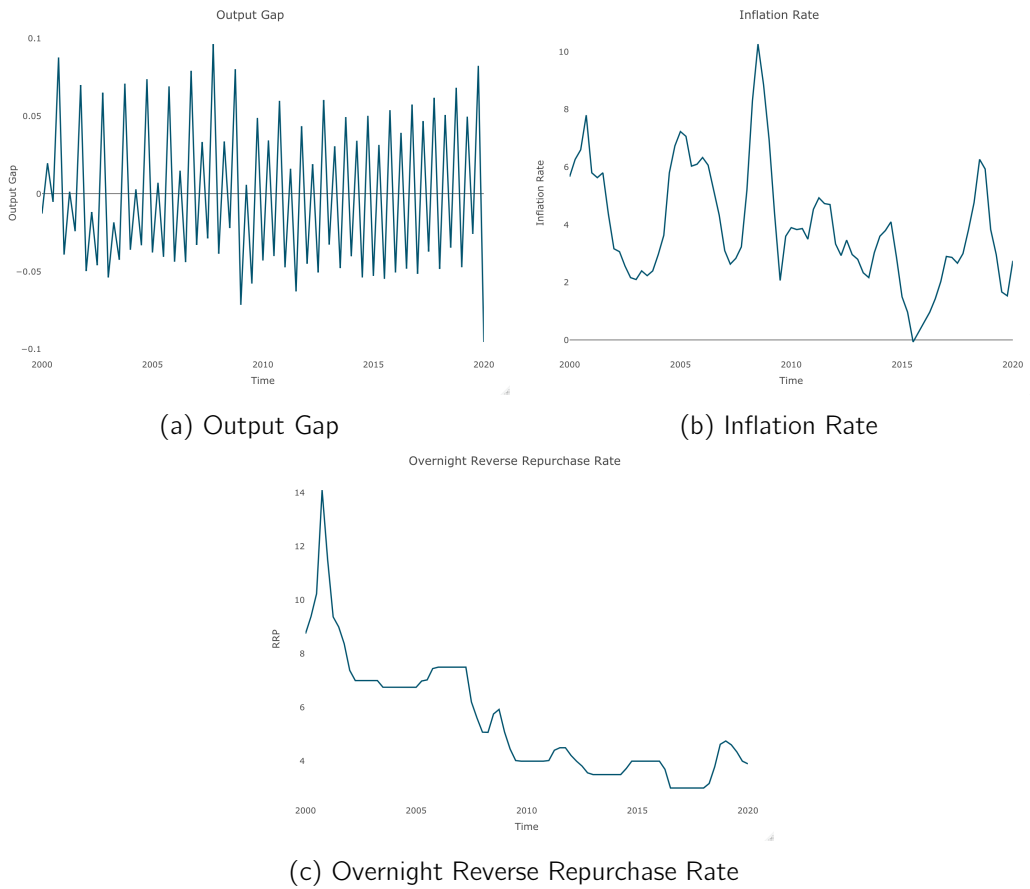


Figure 28: Time Series Plots in SVAR

11.2 Setting Restrictions and Building the SVAR

We will now differentiate from the VAR by adding restrictions. We will need to build the matrix \mathcal{A} and subsequently estimate the structural coefficients by using the recursive ordering method discussed in the last section.

11.2.1 Setting Restrictions

We now come to a crucial part of our analysis in SVAR which is setting the value of matrix \mathcal{A} . As we have said, this matrix is the matrix of contemporaneous shocks affecting the variables in the systems. In identifying the coefficients, we need to set restrictions and these restrictions are defined by an economic principle. In the matrix we will build, we will impose the behavior that if r is the policy rate, we could say that a reaction or a movement in r would be attributable to shocks in π and y in the same period. We will structure the matrix such that the policy rate r only affects π and y with a lag. Hence, we formulate the matrix as the form below.

$$\mathcal{A} = \begin{pmatrix} 1 & 0 & 0 \\ a_{21} & 1 & 0 \\ a_{31} & a_{32} & 1 \end{pmatrix}$$

This restriction suggests that a contemporaneous shock in y affects both π and r in the same period and the lagged values in the system. Further, π shocks only affect r contemporaneously but not y . Likewise, it

also affects the lagged values in the system. Lastly r doesn't affect y and π contemporaneously but affects the lagged values in the system. We first define a matrix as an object "amat" and set the conditions in the code.

```
amat <- diag(3)
amat[2,1] <- NA
amat[3,1] <- NA
amat[3,2] <- NA
amat
```

The `diag()` command creates a 3×3 identity matrix. The next three commands modify the lower triangular to have values of NA. Essentially, we are identifying the parameters which can freely take any value. Hence, after we estimate, the NA will get filled up. Meanwhile, we retain the upper triangular as zero which represents the restrictions we imposed akin to the economic intuition we pre-specified in the matrix.

11.2.2 Binding Variables and Selecting the Cholesky Ordering

After this, we will now specify the ordering we want. To reflect the economic intuition of our restriction, we have to follow the order of output gap first, followed by inflation, and lastly by the policy rate. Again, think of this Cholesky ordering as a timeline of events or a sequencing of events.

```
sv <- cbind(y, pi, r)
colnames(sv) <- cbind("OutputGap", "Inflation", "RRP")
```

11.2.3 Lag Order Selection

We now move on to determining the lag ordering using the `VARselect()` command. Since this is quarterly data, we expect a lag order of around four to six.

```
lagselect <- VARselect(sv, lag.max = 8, type = "both")
lagselect$selection
lagselect$criteria
```

Based on the AIC, SBIC, HQIC, and FPE, it appears that the optimal lag order is five lags. Hence, let us build our model to have 5 lags.

11.2.4 Building the SVAR

To estimate the SVAR, we first need to estimate a reduced form VAR which is why we have a `VAR()` estimation here. Again, we set the number of lags to 5. After which, we use the `SVAR()` command and set the `Amat` option to our `amat` object. Doing this should yield us our estimated SVAR \mathcal{A} matrix.

```
Model1 <- VAR(sv, p = 5, season = NULL, exog = NULL, type = "const")
SVARMod1 <- SVAR(Model1, Amat = amat, Bmat = NULL, hessian = TRUE, estmethod =
c("scoring", "direct"))
SVARMod1
```

If you did this correctly, you should see that we estimated our structural parameters to be 0.21, 0.09, and -0.11 which can be found as the estimated \mathcal{A} matrix.

11.3 SVAR Applications

We will now move on to the applications of SVAR. It is expected that before this, you first diagnose the VAR akin to the steps we used in the last section. In this part, we will focus mainly on two main applications, which are the Impulse Response Functions and the Forecast Error Variance Decompositions.

11.3.1 Impulse Response Functions in SVAR

We will zero in on how the intuition played a role in building the IRFs and seeing how the policy rate r would respond to shocks in the system. To run the IRF, we simply use the `irf()` command and set the impulse and responses as we wish. As there are three variables in the system, there are a total of nine (3^2) possible IRFs to come by. We will zero in on three and I hope you'll understand the reaction clearly based on the restrictions we imposed.

```
SVARog <- irf(SVARMod1, impulse = "OutputGap", response = "OutputGap")
plot(SVARog)
SVARinf <- irf(SVARMod1, impulse = "OutputGap", response = "Inflation")
plot(SVARinf)
SVARrrp <- irf(SVARMod1, impulse = "Inflation", response = "RRP")
plot(SVARrrp)
```

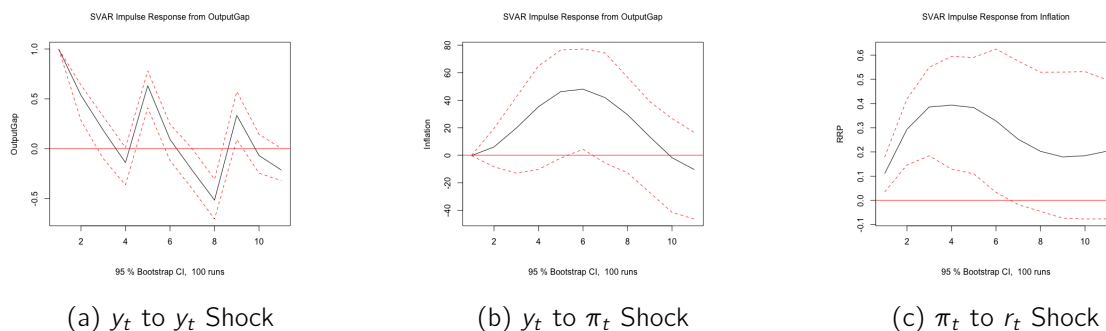


Figure 29: Impulse Responses in SVAR

The impulse responses are very intuitive. If a shock in the output gap starts the sequence, it is expected to increase the output gap. This increase in the output gap increases the inflationary gap which means that the economy is producing more than it was potentially expected to which may cause the economy to overheat. This higher than expected output gap will push inflation upward as productivity continues to increase. Since the central bank sees that output gap increases and that inflation increases, it can accommodate these by increasing the RRP which is exactly what is seen in window (c). This is typically the monetary response of a central bank and was validated by our SVAR.

11.3.2 Forecast Error Variance Decomposition in SVAR

Lastly for this module on VAR and SVAR, we will turn to the FEVD decomposition in the SVAR. To do this, we use the `fevd()` command and set a horizon ahead. In this case, we set it to ten periods ahead. We then use the `plot()` command to plot the values which are seen in figure to follow.

```
SVARfevd <- fevd(SVARMod1, n.ahead = 10)
SVARfevd
plot(SVARfevd)
```

If you will notice, the FEVD of Output Gap is explained purely by Output Gap. This is just because we ordered this one first. If you will see the shocks in the other variables, they reflect that majority of the

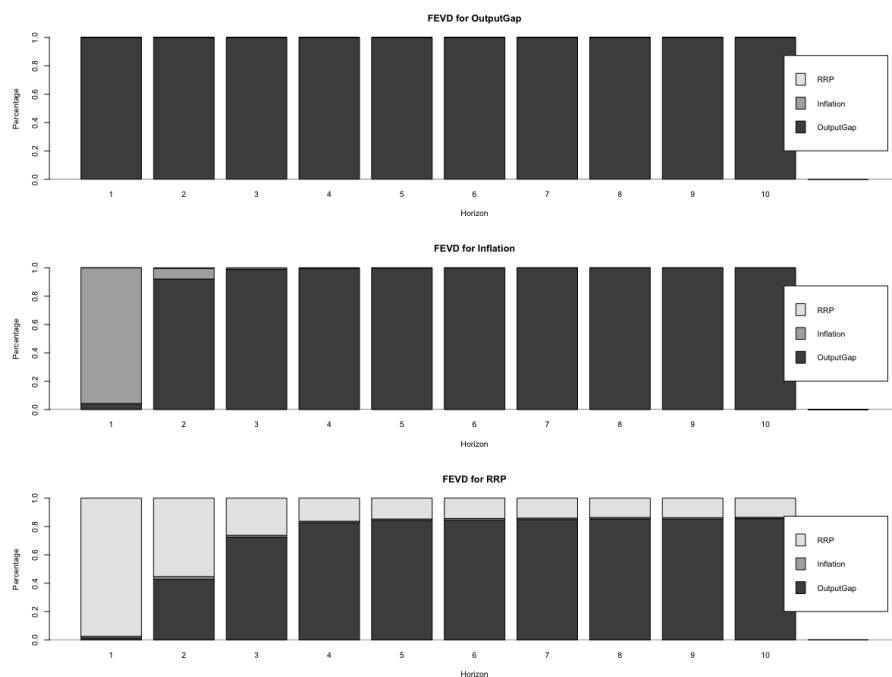


Figure 30: Forecast Error Variance Decomposition in VAR

forecast error variance decomposition was because we pushed that variable as the first in the order.

Overall, I hope you can see how adding a structure to the VAR which are the different restrictions we imposed was able to yield a key policy result. Therefore, I hope you are able to realize that SVARs are incredibly useful policy tools for government in conducting policy and controlling for unobservables.

11.4 Code Summary

```
#SVAR in R
#Justin S. Eloriaga

library(urca)
library(vars)
library(mFilter)
library(tseries)
library(TSstudio)
library(forecast)
library(tidyverse)

#Loading the Dataset

macro <- read_csv(file.choose())
head(macro)

#Creating the Time Series Objectives

y <- ts(macro$'Output Gap', start = c(2000,1,1), frequency = 4)
pi <- ts(macro$CPI, start = c(2000,1,1), frequency = 4)
r <- ts(macro$RRP, start = c(2000,1,1), frequency = 4)

#Time Series Plots

ts_plot(y, title = "Output Gap", Xtitle = "Time", Ytitle = "Output Gap")
ts_plot(pi, title = "Inflation Rate", Xtitle = "Time", Ytitle = "Inflation Rate")
ts_plot(r, title = "Overnight Reverse Repurchase Rate", Xtitle = "Time", Ytitle = "RRP")

#Setting the Restrictions

amat <- diag(3)
amat[2,1] <- NA
amat[3,1] <- NA
amat[3,2] <- NA
amat

#Building the Model

sv <- cbind(y, pi, r)
colnames(sv) <- cbind("OutputGap", "Inflation", "RRP")

lagselect <- VARselect(sv, lag.max = 8, type = "both")
lagselect$selection
lagselect$criteria
```

```
Model1 <- VAR(sv, p = 5, season = NULL, exog = NULL, type = "const")
SVARMod1 <- SVAR(Model1, Amat = amat, Bmat = NULL, hessian = TRUE, estmethod =
c("scoring", "direct"))
SVARMod1

#Impulse Response Functions

SVARog <- irf(SVARMod1, impulse = "OutputGap", response = "OutputGap")
SVARog
plot(SVARog)

SVARinf <- irf(SVARMod1, impulse = "OutputGap", response = "Inflation")
SVARinf
plot(SVARinf)

SVARrrp <- irf(SVARMod1, impulse = "Inflation", response = "RRP")
SVARrrp
plot(SVARrrp)

#Forecast Error Variance Decomposition

SVARfevd <- fevd(SVARMod1, n.ahead = 10)
SVARfevd
plot(SVARfevd)
```

12 Vector Error Correction Model

"Your eyes deceive you, do not trust them" - Obi Wan Kenobi

In training a young Luke Skywalker to be a Jedi, Obi Wan Kenobi uttered the quote above. I think it is relevant now more than ever in an age of social media and rapid advancements in technology and in the way we work and communicate. He made a point in saying that often our most perceptive sense may ultimately cause our fall or end up causing us mistakes. That is why, it is important to trust in the connections, to trust in the *force*. Do not let initial perceptions guide your judgement, for they may deceive you. So what does this have to do with our lesson? Well it plays a lot actually. In multivariate time series, we deal with all sorts of variables with many different characteristics. Some of these variables/series are *stationary*, that is $I(0)$ or integrated of order zero. In other words, these variables need not be differenced to be stationary. On the other hand, there are variables which are *non-stationary* which are integrated of degree 1, $I(1)$, most of the time in economics. This means in order for these variables to be stationary, some transformation like differencing is required.

In this section, we will be focusing on the *Vector Error Correction Model* which is a further deviation like the VAR. In particular, we will talk on the concept of *Cointegration* and the *Error Correction* methodology in addressing the non-stationarity of some time series variables.

12.1 Foundations of Cointegration

To illustrate a classic example of cointegration, let us use three variables to illustrate our point. The three variables are M2 Money Supply, the Consumer Price Index, and real GDP. We will use the natural log of all the variables in this illustration.

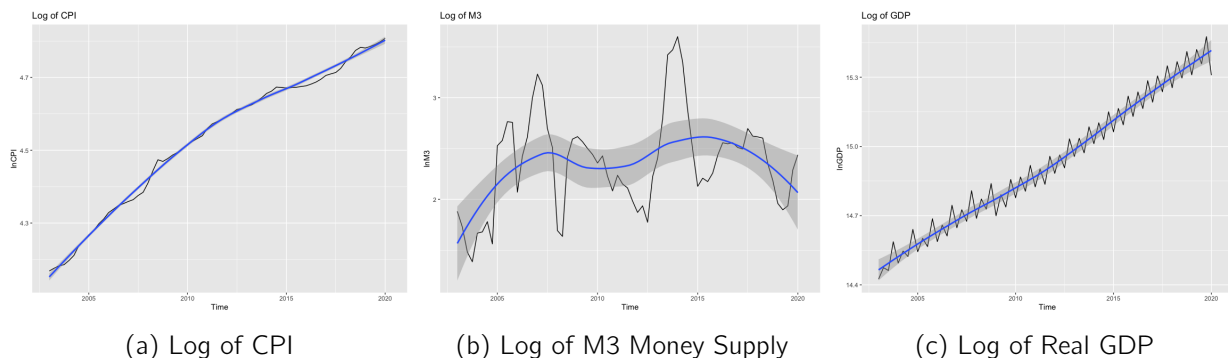


Figure 31: Showing the Series Candidates for Cointegration

We can immediately notice that the series are non-stationary for the three series that were plotted. It is likely that these variables have some upward trend. In discussing cointegration, the key idea is that a series could be non-stationary but at the same time, there could be something deep in these variables that drive two or more of these variables at the same time. In other words, these variables can look non-stationary but at the same time, there could be something that is driving them together. Consider the scatterplots between CPI and GDP and CPI and M3 to follow. Notice that there is a considerable upward association between them. If we ran a regression between them, we would most probably get a positive relationship. We now ask, what could be the underlying motivation or deep reason behind that upward trend or association.

Let us go back to our premise of $I(1)$ and $I(0)$ variables. In general, it is believed that a *linear combination* of $I(1)$ series is also $I(1)$. That is, a linear combination of two or more non-stationary series' will likely yield a non-stationary series.

How does a linear combination look like? Well, essentially, it looks somewhat like this.

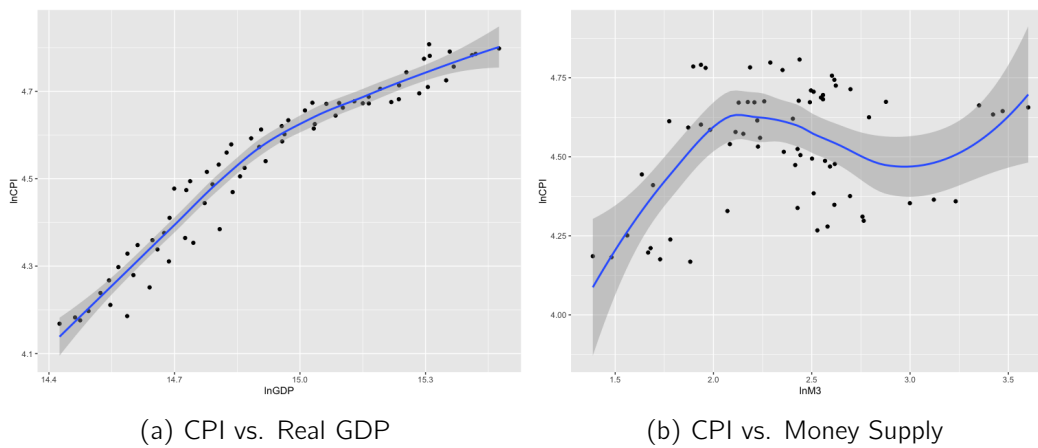


Figure 32: Scatterplots for Cointegration

$$\beta_1 M3 + \beta_2 \text{realGDP} + \beta_3 \text{CPI}$$

Notice that all the β coefficients are raised to one, therefore the expression is linear in parameters. As we said, a linear combination of non-stationary processes most often would yield a non-stationary series too. However, in some special cases, there could be a linear combination which is $I(0)$ or something that is stationary. Something quite special needs to happen for a linear combination to become stationary.

That deep relationship is likely based on economic theory or intuition which could explain as to why some linear combination of $I(1)$ series could be $I(0)$. For example, say we have a commodity in two different markets, namely, the US and the European market. One might think that the prices cannot deviate too much because there would be arbitrage, and of course, traders would move this commodity from one market to another. In other words, the difference between the two processes should not be something non-stationary. It shouldn't have a trend, its variance should be varying.

The concept of *cointegration* is a purely statistical construct. Economic theories can help us understand the intuition and maybe guide us in determining what kinds of variables could be cointegrated. We will now formally define cointegration. An $I(1)$ series is said to be *cointegrated* if there exists at least one linear combination of these variables that is stationary.

12.1.1 How a VAR performs under Cointegration

Of course we now need to move on how to implement this. Since we are dealing with linear combinations of multiple variables, the first approach that comes to mind is a VAR. This is because we are dealing with many variables. The first method which comes to mind is estimating a VAR in levels. But as we said earlier, it could become problematic as the estimation contains $I(1)$ variables. If we run a VAR in levels of variables that are $I(1)$, it is probably that the impulse responses of these variables will tend to not decay because they are $I(1)$. On the other hand, if we are estimating impulse response functions using $I(0)$ variables, they tend to decay. If a small mistake on the parameters is made at the very beginning, then there might be a very big difference in terms of impulse responses. Normally, running VAR on levels means that we lose consistency in medium term and longer term horizons. Using a VAR with $I(0)$ variables at longer horizons makes it more likely that the IRFs will reflect the real IRFs regardless of any error encountered. Running a VAR in levels also suggests that the Granger Causality test will not follow the standard F-test.

12.2 Cointegration and Error Correction

We will now turn to a discussion on the *error correction* mechanism and how it links with what we know so far with cointegration. To do this, let us have an example of an actual economic theory which motivates

cointegration. In particular, let us use our data points in the example above and highlight something deeper on monetary theory. Consider our money market equilibrium condition, which states that money supply (m^s) should be equal to money demand (m^d). In essence, we can represent this identity below.

$$m^s = m^d = \beta_0 + \beta_1 \underbrace{P_t}_{\text{Prices}} + \beta_2 \underbrace{Y_t}_{\text{Income}} + \beta_3 \underbrace{r_t}_{\text{Rate}}$$

We can represent the relationship as a simple regression above. However, in reality, we observe some error term u_t which shouldn't be very persistent and its variance shouldn't rise over time. What we are trying to point out is that the variables P_t , Y_t , and r_t are non-stationary variables for the most part and the theory on money market equilibrium hints at cointegration.

In our example earlier, if $\beta_1 M3 + \beta_2 \text{realGDP} + \beta_3 \text{CPI}$ is stationarity although each series is non-stationary individually, their linear combinations tend to basically move together and tends to be bounded together which suggests that the variables are actually *comoving*. This suggests that there is some long-term trend or path that each series is following. However, in defining a long term path, this beggars the question on potential deviations. We ask the question of what kind of mechanism will ensure that each of these series/variables will react to deviations from this long run relationship or go back to it. There must be some adjustment made by m , p , and y , such that they move together in a manner that deviations from $\beta_1 M3 + \beta_2 \text{realGDP} + \beta_3 \text{CPI}$ remain bounded. The mechanism that will correct for this is known as an *error correction*.

12.2.1 Illustrating an Error Correction

To better understand the concept of an error correction, let us begin with an example. Consider the relationship between the money supply and the prevailing prices. Suppose in the long run, we believe that the economic relationship below is true.

$$m_t = \beta p_t + u_t, \quad \beta > 0$$

Suppose for the moment that there are no error terms u_t and let us study how a deviation from this long run relationship or how this model will come back to this long run relationship after a certain disturbance or deviation. Recall that for an actual cointegration relationship to be true, you need the error term. But, for illustration, we will basically assume no shocks during the period.

Let us first analyze the long run relationship intuitively. If $\beta = 1$, then money is neutral. When we say that money is neutral, it means that changes in the money supply have no effects on real economic variables. This implies that money and prices move exactly in the same way and proportionally. The figure that follows shows the initial long run relationship or link in window a. The blue line represents $m_t = \beta p_t$ which is the long run relationship that we theorize. Now, suppose that there is a sudden increase in money supply m_t such that $m_t - \beta p_t > 0$. We now wonder what would be the dynamics of m_t and p_t such that they come back to the relationship $m_t = \beta p_t$. The dynamics of this are interesting, and is likely related to theory which would then underlie how these economic variables move. You can think of the blue line as a sort of "equilibrium" and any deviation from that is a state not in equilibrium.

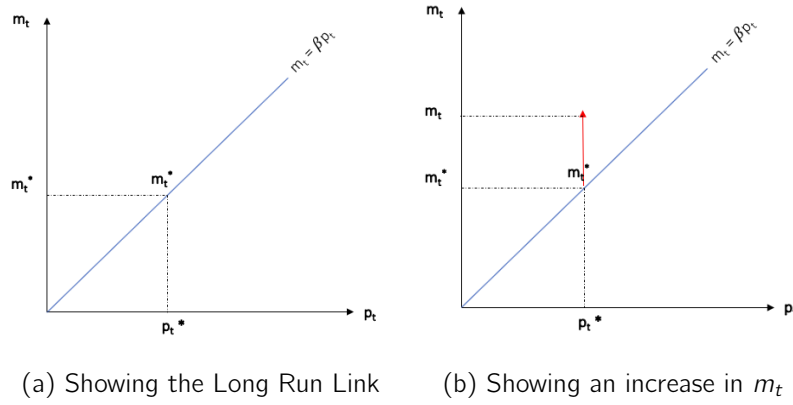


Figure 33: Initial Long Run Relationship and Deviation

There are three possible scenarios that could happen which are illustrated in the figure to follow. First, as seen in window a, the adjustment may be purely due to m_t . Secondly, the adjustment can be purely due to m_t^* . Lastly, it is possible that both p_t and m_t simultaneously adjust. Note that m_t represents the temporary deviation while m_t^* represents the steady state value or the long run value.

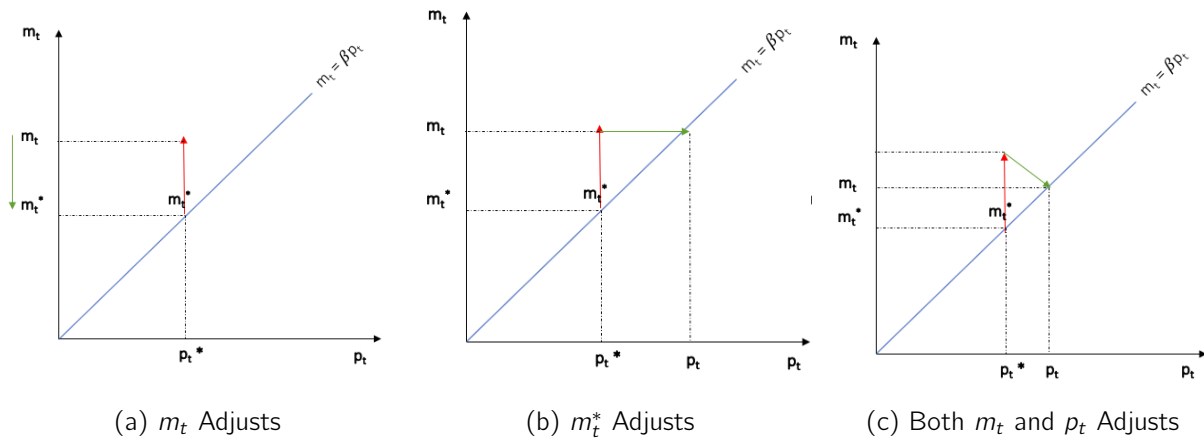


Figure 34: Showing the Error Correction Mechanisms

The first scenario is that m_t is doing all the adjustment. Given that m_t^* increased to m_t (red arrow), one possible adjustment is that m_t^* is unchanged and m_t simply goes down (green arrow).

$$\Delta m_t = \alpha_m (m_{t-1} - m_{t-1}^*), \quad \alpha_m < 0$$

The equation above represents the change in m_t , Δm_t . As we know, the change in m_t back to m_t^* was to decrease m_t . Hence, we can conclude that $\Delta m_t < 0$. On the other side of the equation, we have $(m_{t-1} - m_{t-1}^*)$. Since $m_t > m_t^*$, that entire term is positive. For us to meet the equality, it must be that $\alpha_m < 0$. The α_m is what we refer to as the *error correction coefficient* which is something crucial in our analysis of cointegration. This first case suggests that the intermediate value will adjust back to the steady state value at the original point of equilibrium.

The second scenario represents the adjustment which was purely done by m_t^* . This means that it was the steady state value that reacted to the shock and the shock caused an adjustment to the value in equilibrium. This adjustment subsequently caused prices in equilibrium to increase. Overall, m_t remained unchanged, and p_t and m_t^* both increased.

$$\Delta p_t = \alpha_p (m_{t-1} - m_{t-1}^*), \quad \alpha_p > 0$$

You'll notice that $\alpha_p > 0$, this is because $(m_{t-1} - m_{t-1}^*) > 0$ since m_t is higher than m_t^* . Moreover, we observe that p_t moved to the right (or increased), therefore, $\Delta p_t > 0$. Ergo, it is reasonable to conclude that $\alpha_p > 0$. This is because the error correction affected prices in a positive manner, in that because of the adjustment of m_t^* to the new m_t , p_t subsequently adjusted positively as well. From an economic standpoint, the short run change in p_t is a linear function of the deviation from money neutrality. Referring to window (b), after deviating from this long-run relationship, m_t doesn't move but it's the initial long run m_t^* that goes up to meet m_t . Because of this adjustment, p_t subsequently changes as well.

The last scenario is when both m_t and p_t adjusts which forms the underlying simple error correction model. In window (c), we can see that m_t and p_t are adjusting simultaneously.

$$\Delta m_t = \alpha_m(m_{t-1} - m_{t-1}^*), \quad \alpha_m < 0$$

$$\Delta p_t = \alpha_p(m_{t-1} - m_{t-1}^*), \quad \alpha_p > 0$$

Hence, we notice both of the adjustments seen previously, which is far more realistic. This scenario underlies the construct of error correction. In fact, what we just derived above is referred to as a *basic error correction model*.

12.3 VECM Specification

Given that we now know how a simple error correction works, we will extend the simple error correction more formally by adding back the error term. As we said previously, the error correction mechanism ensures cointegration between a set of variables. Suppose we have a simple error correction model given below.

$$\Delta m_t = \alpha_m(m_{t-1} - \beta p_{t-1}) + e_t, \quad \alpha_m < 0$$

$$\Delta p_t = \alpha_p(m_{t-1} - \beta p_{t-1}) + \varepsilon_t, \quad \alpha_p > 0$$

Running a simple VAR in differences would be *problematic*. This is because it would be miss-specified. A simple VAR in differences would essentially ignore the error correction terms, that is, $\alpha_m, \alpha_p = 0$. This collapses our model into

$$\Delta m_t = e_t$$

$$\Delta p_t = \varepsilon_t$$

This means that the changes in m_t and p_t are just constants, random constants. This, as we know, isn't the case. There could be some defined event or some underpinning relationship which had caused this change. Under this assumption, the simple VAR in differences doesn't match the true dynamics. To somewhat solve this, we can express our error correction model as a *special* VAR. Let us re-express the simple error correction model above by simplifying it. Let's take, for example, $\Delta m_t = \alpha_m(m_{t-1} - \beta p_{t-1}) + e_t$, $\alpha_m < 0$

$$m_t - m_{t-1} = \alpha_m(m_{t-1} - \beta p_{t-1}) + e_t$$

$$m_t = m_{t-1} + \alpha_m(m_{t-1} - \beta p_{t-1}) + e_t$$

$$m_t = (1 + \alpha_m)m_{t-1} - \alpha_m\beta p_{t-1} + e_t$$

Doing the same for $\Delta p_t = \alpha_p(m_{t-1} - \beta p_{t-1}) + \varepsilon_t$, $\alpha_p > 0$, we get

$$p_t - p_{t-1} = \alpha_p(m_{t-1} - \beta p_{t-1}) + \varepsilon_t$$

$$p_t = p_{t-1} + \alpha_p(m_{t-1} - \beta p_{t-1}) + \varepsilon_t$$

$$p_t = \alpha_p m_{t-1} + (1 - \alpha_m\beta)p_{t-1} + \varepsilon_t$$

Therefore, we get the system of equations

$$\begin{aligned}m_t &= (1 + \alpha_m)m_{t-1} - \alpha_m\beta p_{t-1} + e_t \\p_t &= \alpha_p m_{t-1} + (1 - \alpha_m\beta)p_{t-1} + \varepsilon_t\end{aligned}$$

Most of you will have noticed that the system we derived is essentially a $VAR(1)$ system. To be more specific, it is a special VAR system with restrictions all over. In a standard VAR, we would only have β coefficients there. With our special VAR, we have α_m , α_p and a multiplicative combination of α and β . Since we have some of these combinations, these restrictions are *non-linear* by nature. As such, this special VAR has to be estimated with these restrictions in a non-linear manner.

As we mentioned before, there is some sort of economic intuition that implies cointegration between m , p , and y . Hence, there is some belief of the existence of a stationary linear combination of these non-stationary processes.

$$E(b_1 m_{t-1} + b_2 p_{t-1} + b_3 y_{t-1} + \theta) = 0$$

In the expression above, we added some constant θ which just suggests that the linear combination is some stationary process but may exist with some non-zero constant. We can now formally express our *vector error correction model* as the form below. In particular, this is a VECM with zero lags of Δm_t , Δy_t , and Δp_t .

$$\begin{aligned}\Delta m_t &= \alpha_m(b_1 m_{t-1} + b_2 p_{t-1} + b_3 y_{t-1} + \theta) + e_t \\ \Delta p_t &= \alpha_p(b_1 m_{t-1} + b_2 p_{t-1} + b_3 y_{t-1} + \theta) + \varepsilon_t \\ \Delta y_t &= \alpha_y(b_1 m_{t-1} + b_2 p_{t-1} + b_3 y_{t-1} + \theta) + \eta_t\end{aligned}$$

The model above is of lag zero because there are no lags of Δm_t , Δy_t , and Δp_t . Suppose we include one lag, the model will now look like this

$$\begin{aligned}\Delta m_t &= \alpha_m(b_1 m_{t-1} + b_2 p_{t-1} + b_3 y_{t-1} + \theta) + \lambda_{MM}\Delta m_{t-1} + \lambda_{MP}\Delta p_{t-1} + \lambda_{MY}\Delta y_{t-1} + e_t \\ \Delta p_t &= \alpha_p(b_1 m_{t-1} + b_2 p_{t-1} + b_3 y_{t-1} + \theta) + \lambda_{PM}\Delta m_{t-1} + \lambda_{PP}\Delta p_{t-1} + \lambda_{PY}\Delta y_{t-1} + \varepsilon_t \\ \Delta y_t &= \alpha_y(b_1 m_{t-1} + b_2 p_{t-1} + b_3 y_{t-1} + \theta) + \lambda_{YM}\Delta m_{t-1} + \lambda_{YP}\Delta p_{t-1} + \lambda_{YY}\Delta y_{t-1} + \eta_t\end{aligned}$$

A key consideration is how many lags will a VECM model contain. Determining the number of lags starts by running a VAR in levels and determine p . However, when we run a VECM, the order should be $p - 1$. The order is $p - 1$ because if you recall, you can express a VECM as a VAR. However, in doing so, there is a difference of one lag. So, you would need to difference a VAR to obtain a VECM, and in doing so, you basically lose one lag.

12.4 Testing for Cointegration

Of course, in order to say that variables are cointegrated, we would need to test these out. Using our example earlier, let us just add a model intercept for each equation.

$$\begin{aligned}\Delta m_t &= \mu_m + \alpha_m(b_1 m_{t-1} + b_2 p_{t-1} + b_3 y_{t-1} + \theta) + \lambda_{MM}\Delta m_{t-1} + \lambda_{MP}\Delta p_{t-1} + \lambda_{MY}\Delta y_{t-1} + e_t \\ \Delta p_t &= \mu_p + \alpha_p(b_1 m_{t-1} + b_2 p_{t-1} + b_3 y_{t-1} + \theta) + \lambda_{PM}\Delta m_{t-1} + \lambda_{PP}\Delta p_{t-1} + \lambda_{PY}\Delta y_{t-1} + \varepsilon_t \\ \Delta y_t &= \mu_y + \alpha_y(b_1 m_{t-1} + b_2 p_{t-1} + b_3 y_{t-1} + \theta) + \lambda_{YM}\Delta m_{t-1} + \lambda_{YP}\Delta p_{t-1} + \lambda_{YY}\Delta y_{t-1} + \eta_t\end{aligned}$$

We can write the equations in our VECM in matrix form by stacking the equations and variables into some vector X_t to obtain the matrix equation

$$\Delta X_t = \Gamma + \Pi X_{t-1} + \Lambda \Delta X_{t-1} + u_t$$

In this VECM, Γ represents a vector of constants, ΠX_{t-1} is a matrix of error correction terms, $\Lambda \Delta X_{t-1}$ is a matrix of the different first order lags and u_t is a vector of error terms. This is reminiscent of the augmented Dickey Fuller test.

12.4.1 Johansen's Methodology

Johansen's methodology is based on the idea that estimating the rank of Π gives us information about whether there is cointegration and the number of these cointegrating relationships. By definition, the rank of Π is the maximum number of independent vectors within this matrix. If we have three endogenous variables, we can only have three independent vectors and no more than that. The rank could be zero or at most three or any where in that range. That is, the rank cannot exceed the number of endogenous variables in the system.

If the rank of Π is equal to zero, then it is said that there is *no cointegration*. This implies that $\Pi = 0$ and you can see this in the equation. If you substituted that in the equation, then all the error correction terms would disappear, hence the system collapses to a simple VAR in differences. If the rank of Π is equal to the number of endogenous variables which we refer to as a *full rank*, then it suggests that all variables are $I(0)$ or the variables are all stationary. So in that case, we cannot talk about cointegration between non-stationary variables as the variables are already stationary. Hence, you can just run a VAR in levels. If the rank of Π is strictly between zero and the number of endogenous variables, $0 < \rho(\Pi) < n$ or less than a full rank, then there are r number of independent cointegrating relationships where $r = \rho(\Pi)$.

There are two types of tests in the Johansen methodology. The first is the *trace statistic* for each rank r . In the case of our model with three endogenous variables, r being 0, 1, or 2. The null hypothesis of this statistic is that the rank is at most r versus the alternative hypothesis that the rank is strictly greater than this r . The second is the *maximum eigenvalue statistic* for each rank r . The null hypothesis is that the rank is equal to $\rho(\Pi)$ versus the alternative hypothesis is that the rank is $\rho(\Pi) + 1$.

12.5 Johansen Cointegration in R

The time has come to provide an actual example testing for cointegration in R. To do this, we need to use the "urca" package which contains the `ca.jo` command to implement the test. To start of, we load the packages required and load the dataset "VECM_LectureNotes.csv" using the `read_csv()` command.

```
library(urca)
library(forecast)
library(tidyverse)

data <- read_csv(file.choose())
head(data)
```

We then declare our time series variables using the `ts()` command. We note that the series starts in Q1 of 2001 until Q1 of 2020. This is obviously quarterly data so we set the frequency to four.

```
GDP <- ts(data$lnGDP, start = c(2003,1,31), frequency = 4)
CPI <- ts(data$lnCPI, start = c(2003,1,31), frequency = 4)
M3 <- ts(data$lnM3, start = c(2003,1,31), frequency = 4)
```

After declaring the series as a `ts` object, we will then need to bind the variable into a singular system. To do this, we use the `cbind()` command. We store the said system into an object, in this case `dset`.

```
dset <- cbind(GDP,CPI,M3)
```

After which, we would need to select the optimal lag order. Now recall that in a VECM (or the presumption of cointegration), the optimal lag order to use is $p - 1$. To determine the optimal number of lags, we use the `VARselect()` command.

```
lagselect <- VARselect(dset, lag.max = 7, type = "const")
lagselect$selection
```

Since 5 came up in the Hannan Quinn and the Final Prediction error, we choose to use 4 lags. 4 lags is expected given that the frequency of the data is quarterly. Now that we have our lag selection done, we can move on to specifying the test. As we mentioned, there are two variants of the Johansen test. These are the maximum eigenvalue and the trace statistic approach. Conveniently, the `ca.jo()` command allows us to specify these with ease.

We start off with the trace statistic approach. Again, the `ca.jo()` command implements the Johansen cointegration methodology on our system (`dset`). We use the "trace" statistic approach specified as the type in this case. Next, we specify that our specification has a constant (`ecdet = "const"`) and that we are using 4 lags.

```
ctest1t <- ca.jo(dset, type = "trace", ecdet = "const", K = 4)
summary(ctest1t)
```

If you ran the test properly, you should see the table below.

	test	10pct	5pct	1pct
$r \leq 2$	7.89	7.52	9.24	12.97
$r \leq 1$	21.64	13.75	15.67	20.20
$r = 0$	87.77	19.77	22.00	26.81

Figure 35: Johansen Cointegration Result

In the table, you should see 4 columns. The test column contains the test statistics, while the three other columns contain the critical values at a 10 percent, 5 percent, and a 1 percent level. For this case, as standard practice, we often use the 5 percent critical value as reference. The r in the table represents the rank and we know that this is some indication of the number of cointegrating relationships. When $r = 0$, the test stat $87.77 > 22$. This means that we reject the null hypothesis which suggests that $r > 0$. As such, there is some cointegration present. When $r \leq 1$, $21.64 > 15.67$. This again means that we reject the null hypothesis which suggests that $r > 1$. Lastly, when $r \leq 2$, we fail to reject the null hypothesis because $7.89 < 9.24$. Therefore, we conclude that there are at most 2 cointegrating relationships present.

We can also run the test using the maximum eigenvalue variant using the commands below.

```
ctest1e <- ca.jo(dset, type = "eigen", ecdet = "const", K = 4)
summary(ctest1e)
```

As we can see, we have similar results as with the trace approach. Overall, we can conclude that there are at least two cointegrating relationships present. By now, you will have connected the dots that these two relationships are likely prices and the money supply and prices and GDP. We will get into that later on once we formally estimate a VECM.

12.5.1 Code Summary

```
#Testing for Cointegration using the Johansen Test
#Justin S. Eloriaga

library(urca)
library(forecast)
library(tidyverse)

#Loading the Dataset

data <- read_csv(file.choose())
head(data)

#Declare the Time Series Objects

GDP <- ts(data$lnGDP, start = c(2003,1,31), frequency = 4)
CPI <- ts(data$lnCPI, start = c(2003,1,31), frequency = 4)
M3 <- ts(data$lnM3, start = c(2003,1,31), frequency = 4)

#Creating our System

dset <- cbind(GDP,CPI,M3)

#Selecting the Optimal Number of Lags (Recall, this is p - 1)

lagselect <- VARselect(dset, lag.max = 7, type = "const")
lagselect$selection
lagselect$criteria
#Since 5 came up the most, we use (5-1) or 4 lags

ctest1t <- ca.jo(dset, type = "trace", ecdet = "const", K = 4)
summary(ctest1t)

ctest1e <- ca.jo(dset, type = "eigen", ecdet = "const", K = 4)
summary(ctest1e)

#Hence, we have two cointegrating relationships in this model
```

13 Course Problem Sets

The problem sets are answerable in around seven to eight hours each. They involve a mix of objective essay, proving, and practical application through simulation or regression in R. All problem sets are done *individually* and emailed to justin.eloriaga@dlsu.edu.ph by the Friday of the respective weeks of submission.

Each problem set is worth 30 percent of your grade, so make it count. All answers should be compiled in one Word document and converted to pdf for submission. Copy the code to the document if applicable and paste figures from R. For all proofs, use equation editors like Word's build in Equation Generator or MathType to generate the equations. You may also use L^AT_EX if you prefer. For all objective essays, type this out in a clear yet concise manner. You may make a table if it is asking you to compare and contrast. For all practical applications, you are always requested for intuitive answers with their corresponding implications. As such, start with a brief overview of what you are doing, paste the accompanying results or graphs, and interpret. Any code used for a particular item should be pasted in the Word Document after that item has been fully answered.

For you to have an easier time deliniating which question type, the label codes are provided below.

- **PD** Proving or Deriving
- **OE** Objective Essay
- **A** Practical Application

To make it easy for the students to target a grade, each problem set comprises of only ten numbers each. You chose ten items from a question bank consisting of twelve questions. We will follow an equally divided system wherein each question has the same weight. Having ten questions a problem set suggests that each item is worth three percent of your grade. There are some items that take only 1 minute to answer while there are some that take roughly twenty to thirty minutes. This allows you to better pick your battles. However, notice that a heavy bias is given to practical applications as these problems will enable to you apply all concepts in a useful manner and will train you to become more confident in coding.

Rush not into fights, longs the war. Only by surviving it, will you prevail - Yoda

As the great and wise Yoda once said the quote above, let us heed his advice. Take your time to really learn the concepts. We are after mastery more than sheer breadth. I highly suggest for you to target answering one to two problems every week. This will spread out your work and give you the best chance of success. It is not recommended that you "wing-it" it and answer everything before the deadline. Try not to procrastinate.

As stated in the syllabus, you need to submit electronically a pdf version of your word-processed assignment. PLEASE NAME the pdf file SURNAME_ECO602M_ProblemSetX. The deadline for electronic submission of the pdf copy is the Friday of the week of submission. Below are the deadlines for the submission. You are also required to submit the .pdf copy of the submission in AnimoSpace.

Problem Set 1: Week 7

Problem Set 2: Week 13

For any queries, merely use the lecture notes as well as the videos on YouTube. You will need R to accomplish this but if you are more comfortable in Eviews or Python, be my guest. For any more personal inquiries, merely send me a private email and I will get back to you as soon as possible.

13.1 Problem Set 1 - Univariate Time Series

1. **OE** Highlight the main differences between an AR process and an MA process. How can we possibly distinguish the two processes and tell them apart?
2. **OE** What are the key differences between the ACF and the PACF of an AR and MA process. What may be the case for this?
3. **PD** Transform the $AR(p)$ model as an $MA(\infty)$ process. What is the intuition behind this transformation?
4. **PD** Prove that a simple random walk $y_t = y_{t-1} + u_t$ is not a stationary process.
5. **PD** Consider the $AR(2)$ model $y_t = 2y_{t-1} - y_{t-2} + u_t$. Derive the characteristic equation of this model and prove that the model is non-stationary. What may be the cause of this non-stationarity?
6. **PD** Consider an $ARMA(2, 2)$ model given as $y_t = \mu + \theta_1 u_{t-1} + \theta_2 u_{t-2} + \phi_1 y_{t-1} + \phi_2 y_{t-2} + u_t$. First, generate the scalar forecasts up to three periods ahead. Secondly, generate the conditional (actual) forecasts up to three periods ahead. Lastly, what can be inferred about the forecasts greater than 3? Why may this be the case?
7. **A** Simulate an $AR(3)$ and an $MA(3)$ process in R using the `arima.sim()` command to do so. Set the AR coefficients and the MA coefficients to any value below 1. Generate the ACF and PACF of the said processes and verify the properties highlighted in the comparison table.
8. **A** Simulate two $AR(1)$ process in R with $\phi = 0.99$ in one and $\phi = 0.10$ in the other. Perform an Augmented Dickey Fuller Test, a Phillips Perron Test, and a KPSS test on each model. What do you observe? Is this reasonable?
9. **PD** By definition, the mean squared error (MSE) is a combination of bias and the standard forecast error (SE). Prove that the identity $MSE = SE^2 + BIAS^2$ holds. You may use a direct or indirect proof.
10. **A** In the folder, there is a file called *WeeklyExchangeRate*, which, by the name of the file, is the Weekly USD-PHP exchange rate. Generate two in-sample forecasts, one using an $ARIMA(1,1,1)$ and the other using the specification most preferred by the `auto.arima()`. Which forecast is better? Why may this be the case? Using which model you this is better, generate an out of sample forecast for 8 weeks ahead. What was the reported value seven weeks ahead?
11. **A** In the folder, you should see the average adjusted weekly closing stock price of three entities, namely, SMDC, 2GO, and the PSEi index. Generate a forecast for each model one step ahead (blind forecast) using any model you deem most fit. You may conduct in-sample testing to verify the accuracy of your model. Say I gave you ₱1,000,000 pesos today and you had to invest it all in either SMDC or 2GO, which one would you invest it in based on the results of your one-step ahead forecast (Note, this is the one that gives the higher profit or the lower loss). What is your expected profit (or loss) in ₱? Can I say that my investment in either 2GO or SMDC beat the market (i.e. PSEi)? *(This item is worth 5 percent, as such, a bonus of 2 percent is embedded should this be answered well)*
12. **A** Using the same codes and the same dataset as in our *Inflation* example, use any other model apart from the three models used. Use your creativity to generate a forecasting model which outperforms at least model 1. Brownie points will be given if the model generated is better than the model generated using `auto.arima()`. After which, forecast out-of-sample six months ahead. *(This item is worth 5 percent, as such, a bonus of 2 percent is embedded should this be answered well)*

13.2 Problem Set 2 - Multivariate Time Series

1. **OE** Highlight using a table or bullets the key similarities and differences between the VAR, SVAR, and VECM methodologies. What innovation or assumptions did each one push for and that defined its methodology? What were the key assumptions?
2. **PD** Derive the $VMA(\infty)$ from the $VAR(p)$ process. What key intuition/s can we obtain from the $VMA(\infty)$ form.
3. **A** Using the same data in the Example on VAR which used the study of Christopher Sims, estimate the entire model but recalibrate it to use 14 lags, similar to the specification of Christopher Sims. What can you observe? What were the similarities and differences?
4. **A** Using the same data in the Example on VAR which used the study of Christopher Sims, recall that we encountered a *puzzling result*. Sims solved this result by including commodity prices and the log of the exchange rate to alleviate the puzzling result. Using the same codes, modify the model by adding the log of the USD/Php exchange rate and a commodity price index following the order RRP, log of Exchange Rate, Commodity Prices, M1, CPI, and InIP. Show that the puzzle had been solve using your estimated IRFs. Note, please get the data yourself and incorporate it in the model. Just search online and they should be there. *(This item is worth 5 percent, as such, a bonus of 2 percent is embedded should this be answered well)*
5. **A** Using the dataset on Okun, estimate a bivariate VAR and perform all steps outlined in the Example of VAR section of the lecture notes. What are the key conclusions we can draw?
6. **A** Using the dataset on Okun, estimate a bivariate SVAR. Impose the restriction which you believe is adept and explain your reasoning. What are the key conclusions we can draw?
7. **A** Using the dataset on Okun, estimate a bivariate VECM. Is it apt that VECM is used based on the number of cointegrating relationships?
8. **A** Using the StockPricesVECM.csv file, determine the number of cointegrating relationships among the system of three stock prices. Use the Johansen Trace Statistic approach and the Maximum Eigenvalue approach. Do these tests agree with each other?
9. **OE** Explain the process of an error-correction and why this mechanism is important in ensuring cointegration.
10. **A** Using APPL.csv, forecast the stock price of Apple 6 periods ahead using a simple LSTM model. Set the epoch equal to 1000. Compare this forecast with a simple `auto.arima()` out of sample forecast. Which is better? *(This item is worth 5 percent, as such, a bonus of 2 percent is embedded should this be answered well)*
11. **A** Go to Medium.com and follow any article that uses a Support Vector Autoregression (not to be confused with Structural Vector Autoregression). Using the same dataset as in the lecture notes example, compare the out of sample forecasts of the Support Vector Autoregression and the Structural Vector Autoregression. What do you see? Note, you may need to generate the forecasts for the structural vector autoregression first. *(This item is worth 5 percent, as such, a bonus of 2 percent is embedded should this be answered well)*
12. **A** Using the Model Building in Eviews, generate a forecast for the impact on consumer expenditure of a decrease of 30 percent in tourism arrivals in the country (total). *(This item is worth 5 percent, as such, a bonus of 2 percent is embedded should this be answered well)*

14 Supplementary Code Appendix

Scatterplots using ggplot

Use the dataset on welfare and income which is a cross section of countries from the World Bank. Link to the dataset is in the description box of the YouTube video on "ggplot for Economics" in my channel.

```
library(tidyverse)

data <- read_csv(file.choose())
data

m <- ggplot(data, aes(lnGDPPC, lnConsumption, color = IncomeGroup, shape = Region)) +
  geom_point() + geom_abline() + geom_text(label = CC)

print(m + ggtitle("Income and Welfare") + labs(y="Log GDP per capita, 2017", x = "Log
Consumption per capita, 2017") + labs(colour = "Income Group"))

s <- ggplot(data, aes(lnConsumption, lnGDPPC, label = CC, color = IncomeGroup)) +
  geom_text(size = 2, hjust = 0, nudge_x = 0.033) + geom_point() + geom_abline()

print(s + ggtitle("Income and Welfare") + labs(y="Log Consumption per capita, 2017", x = "Log
GDP per capita, 2017") + labs(colour = "Income Group")) + theme(legend.position = c(0.87,0.2))
```

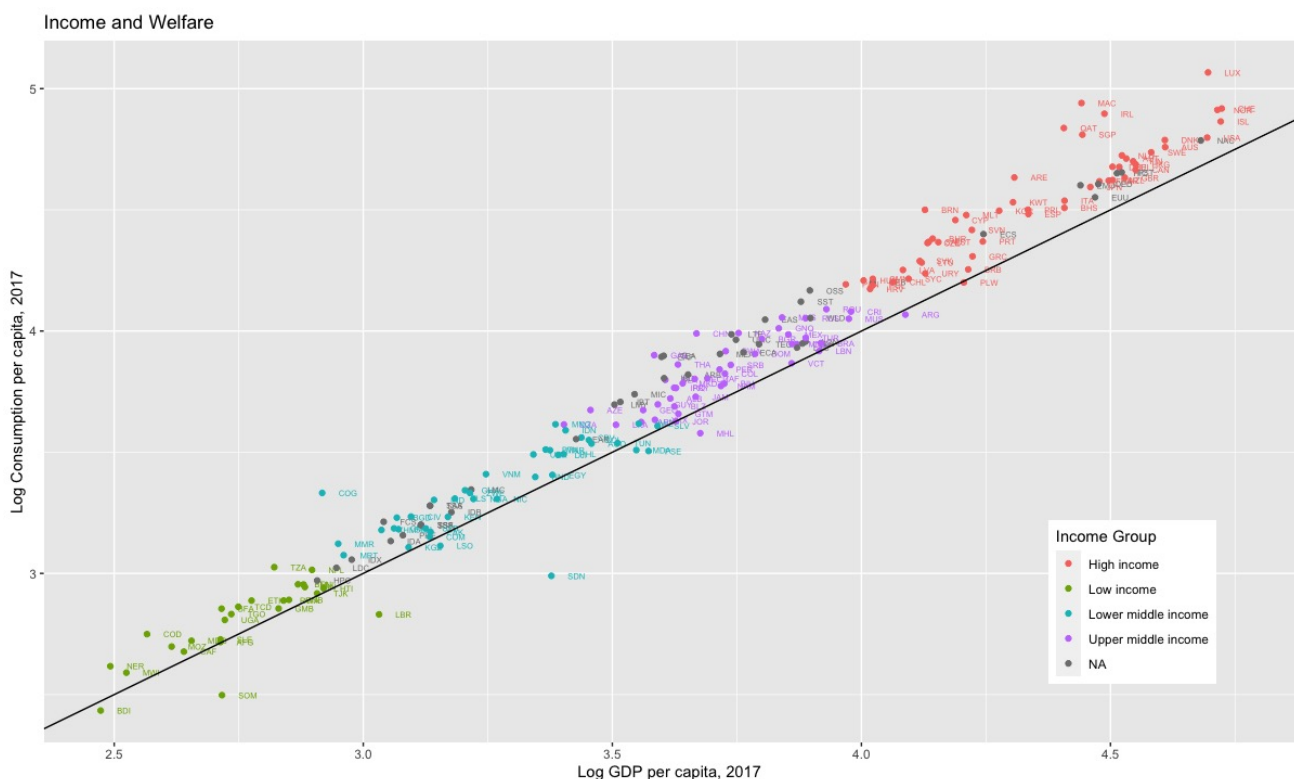


Figure 36: Final Scatterplot

Converting Daily, Weekly, or Monthly Data to Quarterly Data

Credits to Ms. Ellysah Guliman for teaching me this.

```
data <- read_csv(file.choose())
data$Date = as.Date(data$Date, format = "%m/%d/%y")
data <- arrange(data, Date)
data$qdate <- as.yearqtr(data$Date)
data_qtrly <- data %>%
  group_by(qdate) %>%
  summarise_all(mean)
write_csv(data_qtrly, file = "ConvertedFile.csv", row.names = TRUE, fileEncoding = "UTF-8")
```

Plots in VECM

To replicate, use the dataset VECM_LectureNotes.csv.

```
library(tidyverse)

cointeg <- read_csv(file.choose())
head(cointeg)

lnGDP <- ts(cointeg$lnGDP, start = c(2003,1,31), frequency = 4)
lnCPI <- ts(cointeg$lnCPI, start = c(2003,1,31), frequency = 4)
lnM3 <- ts(cointeg$lnM3, start = c(2003,1,31), frequency = 4)
lnConsumption <- ts(cointeg$lnConsumption, start = c(2003,1,31), frequency = 4)

autoplot(lnGDP) + geom_smooth() + ggtitle("Log of GDP") + labs(xlab = "Time", ylab = "lnGDP")
autoplot(lnCPI) + geom_smooth() + ggtitle("Log of CPI") + labs(xlab = "Time", ylab = "lnCPI")
autoplot(lnM3) + geom_smooth() + ggtitle("Log of M3") + labs(xlab = "Time", ylab = "lnM3")
autoplot(lnConsumption) + geom_smooth() + ggtitle("Log of Consumption") + labs(xlab = "Time", ylab = "lnConsumption")

ggplot(cointeg, aes(lnM3, lnCPI)) + geom_point() + geom_smooth() + labs(xlab = "lnM3", ylab = "lnCPI")
ggplot(cointeg, aes(lnConsumption, lnCPI)) + geom_point() + geom_smooth() + labs(xlab = "lnConsumption", ylab = "lnCPI")
ggplot(cointeg, aes(lnGDP, lnCPI)) + geom_point() + geom_smooth() + labs(xlab = "lnGDP", ylab = "lnCPI")
```


Forecasting using Facebook Prophet Libraries

To a person with little or no forecasting knowledge or to some business analyst who just wants to see a simple trajectory, the use of a Facebook developed library called *prophet* is very useful. We use the file *InflationRateProphet.csv*. Note that all we changed here is the name of the date column to *ds* and the series column to *y*. Note that this works best using daily data but we will use our existing monthly frequency for consistency in example. Prophet works best for data with seasonality and clear time effects, which is very useful in business applications.

```
#Forecasting using Prophet in R
#Justin Raymond S. Eloriaga

#Loading the Packages

install.packages('prophet')
library(prophet)
library(tidyverse)

#Loading the Dataset

inf <- read.csv(file.choose())
head(inf)

#Calling the Prophet Function to Fit the Model

Model1 <- prophet(inf)
Future1 <- make_future_dataframe(Model1, periods = 12)
tail(Future1)

Forecast1 <- predict(Model1, Future1)
tail(Forecast1[c('ds','yhat','yhat_lower','yhat_upper')]))
plot(Model1, Forecast1)

#Plotting the Model Estimates

prophet_plot_components(Model1, Forecast1)
dyplot.prophet(Model1, Forecast1)
```

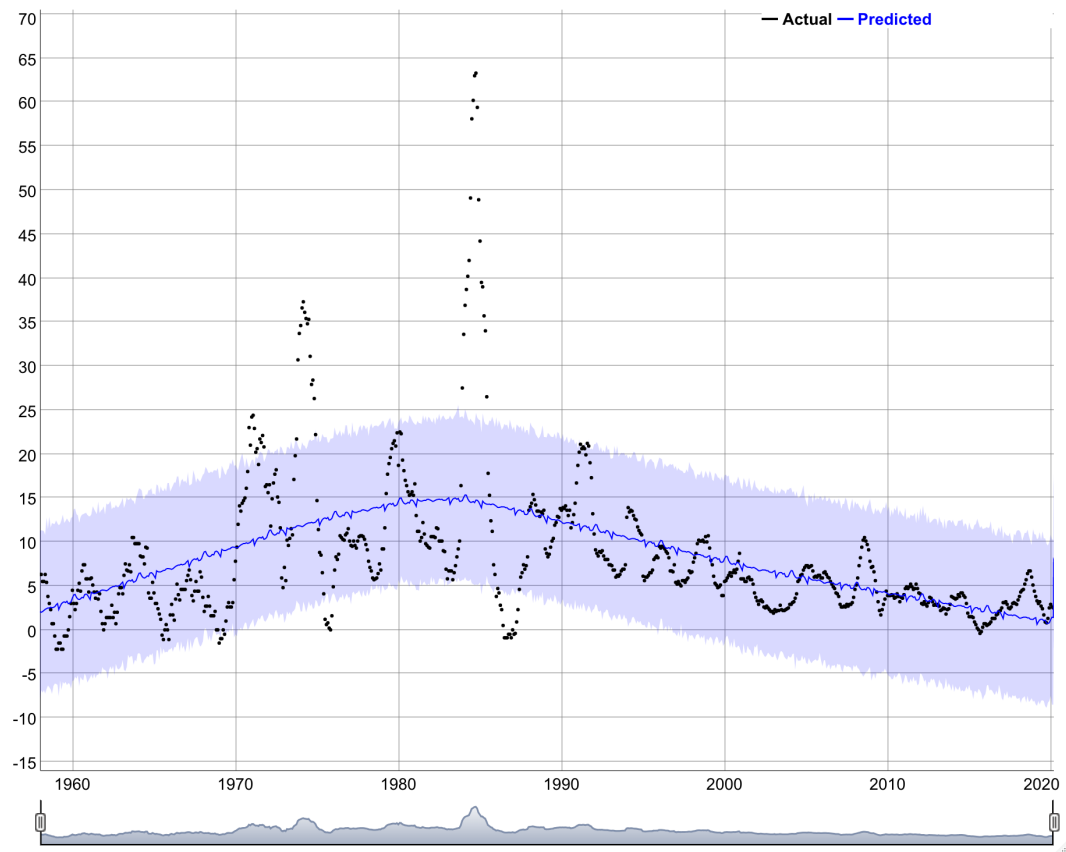


Figure 37: Prophet's Forecast

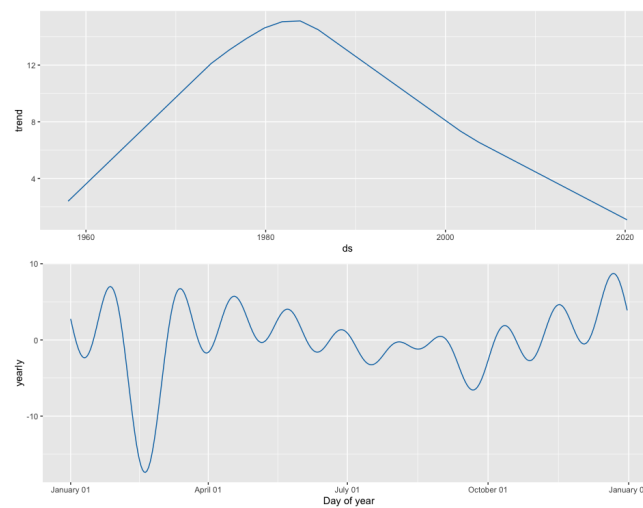


Figure 38: Series Decomposition

15 Course Syllabus

Course Details

Course Title	Time Series Econometrics (ECO602M)
Prerequisite	None
Prerequisite to	ECO602W
Faculty	Justin Raymond S. Eloriaga
Term/Time/Room	Term 3 AY 2019 – 2020 / 18:00 – 21:15 (G01) Wednesday / Full Online

Course Description

This course is a rigorous introduction to the fundamentals of time series econometrics. In particular, this course aims to introduce students to the theoretical and empirical underpinnings behind the classical and modern forecasting methodologies with various applications to microeconomics and macroeconomics. This course starts with an introduction to the stochastic processes and the notion of a time series, differentiating non-stationary and stationary data and discussing the different classes of univariate stationary forecasting models such as the Autoregressive, Moving Average, and Autoregressive Moving Average. After which, the course shall generalize these constructs to the multivariate forecasting models such as the Vector Autoregression, Structural Vector Autoregression, and Vector Error Correction Models. Other topics such as advanced panel data models, non-parametric forecasting techniques, and other forecasting methods may also be dabbled into from time to time, depending on the progression of the class. These fall into the Brown Bag sessions outlined in the course timeline.

This course relies heavily on concepts that have been introduced in economic statistics and basic econometrics such as regression, probability density functions, hypothesis testing, and key classical linear regression model assumptions. Therefore, students enrolled in the course are expected to review their materials from the previous classes as these will be very helpful in understanding some of the ideas in this course. Excellent resources to review the necessary prerequisites are Brooks (2019) and Wooldridge (2016).

Course Objectives

This course is intended to achieve the following

1. Familiarize students with the classical and modern forecasting models that have shaped how we analyze economic markets and control economic quantities for the purposes of projection or scenario analysis.
2. Introduce students to univariate and multivariate forecasting techniques with applications to forecasting in microeconomics and macroeconomics.
3. Inculcate students with the skills necessary to comprehend and perform the state-of-the art econometric methodologies.
4. Enhance students' problem-solving, critical thinking, and analytical skills by using verbal reasoning, graphics, statistics, regression, and mathematics to evaluate economic problems and issues.

Course Learning Outcomes

See the table entitled Course Learning Outcomes. During the course, students are expected to improve their written communication, interpersonal communication, problem solving, numeracy, and teamwork skills. Students are also expected to develop their skills in computer programming and working with a programming language, namely, R. The course will try and introduce the language as easily and intuitively as it can. Students however are expected to review codes and explore other coding practices which may further advance their study.

Learning Outcome (ELGA)	Learning Outcomes
Intellectually Inquisitive	LO1: Identify the basic time series process by differentiating stochastic from deterministic, static and dynamic, stationary from non-stationary. LO2: Distinguish between the most fundamental stationary univariate forecasting approaches. LO3: Extend basic concepts to multivariate forecasting and relate necessary economic intuition whether in a structural or non-structural manner. LO4: Identify the best forecasting or econometric model to be used given a particular set of conditions, assumptions, or limitations. LO5: Apply basic forecasting techniques to model economic quantities and forecast in-sample.
Technically Proficient	LO6: Apply forecasting techniques to perform out of sample forecasts using a plethora of models with varying assumptions and results. LO7: Model changes in the economy by introducing innovations and disturbances. LO8: Functionally articulate forecasts and simulations in a palatable manner.
Agent of Positive Social Change	LO9: Identify key economic development issues where counterfactual scenarios may be implemented to determine the cost of policy.
Globally Competitive	LO10: Explain in non-technical terms the essence of forecasting and communicate results in a palatable manner.

Learning Plan

The term is divided into fourteen weeks. An estimate of the topics covered per week are given in the table to follow. This will closely follow the flow in the lecture notes. As such, it is expected that the student continually refer to the notes in case they miss sessions or need to refresh on certain topics. Each week, we will meet synchronously for 45 minutes to 1 hour and 15 minutes. This time will mostly be used for the discussion of important concepts and sharing and debating paper topics. After this synchronous session, students are expected to watch the YouTube lectures on my channel and attempt answering the Problem Sets. Use the repeat button on the videos *liberally*. The remaining time shall also be used for any queries regarding lessons.

Week	Topic	Learning Activities
0	<i>Review of Basic Econometrics</i>	Class Discussion
1	<i>Introduction to Time Series</i>	Problem Set 1 Term Paper Class Discussion
2	<i>Moving Average and Autoregressive</i>	Problem Set 1 Term Paper Class Discussion
3	<i>Naïve Forecasting and Forecasting Errors</i>	Problem Set 1 Term Paper Class Discussion
4	<i>Box Jenkins Methodology</i>	Problem Set 1 Term Paper Class Discussion
5	<i>Forecast Evaluation and Application</i>	Problem Set 1 Term Paper Class Discussion
6	<i>Vector Autoregression (VAR)</i>	Problem Set 2 Term Paper Class Discussion
7	<i>Structural Vector Autoregression (SVAR)</i>	Problem Set 2 Term Paper Class Discussion
8	<i>Cointegration and Error Correction</i>	Problem Set 2 Term Paper Class Discussion
9	<i>Vector Error Correction Model (VECM)</i>	Problem Set 2 Term Paper Class Discussion
10	<i>Brown Box Topic 1</i>	Problem Set 2 Term Paper Class Discussion
11	<i>Brown Box Topic 2</i>	Problem Set 2 Term Paper Class Discussion
12	<i>Course Capstone and Model Combination</i>	Problem Set 2 Term Paper
13	<i>Research Break</i>	Class Discussion
14	<i>Group Presentations</i>	Term Paper Paper Presentation

Assessment and Requirements

The requirements and their respective grade allocations are discussed in this section. The details of the assessments are also given.

Requirements and Breakdown

Student Assessment Items	Due Date	Weighting	Learning Outcomes
<i>Problem Set 1</i>	Week 7	30%	LO1, LO2, LO5, LO6, LO8
<i>Problem Set 2</i>	Week 13	30%	LO1, LO2, LO3, LO4, LO5, LO6, LO7
<i>Empirical Paper</i>	Week 14	40%	LO1, LO2, LO3, LO4, LO5, LO6, LO7, LO8, LO10

Grading System

96 – 100.0	4.0
90 – 95.99	3.5
84 – 89.99	3.0
78 – 83.99	2.5
72 – 77.99	2.0
66 – 71.99	1.5
60 – 65.99	1.0
Below 60	0.0

Remember that all graduate courses require a minimum credit of 2.0 and all undergraduate courses require a minimum credit of 1.0. Your grades are rounded *off*, not rounded *up* to the nearest two decimal place.

Details of the Assessments

Problem Sets Students will be given problem sets that tackle the application of the concepts and techniques that are currently being discussed in class or to be discussed soon. Students are expected to have read the appropriate references before tackling each problem set. These assignments are an individual effort. You are not allowed to consult with nor be consulted by your classmates. A grade of 0.0 will be given for reports that are suspected to be copies (in full or in part) of each other.

Your answers should be word processed (MSWord or Pages or \LaTeX) and printed on A4-size or short bond papers (front side only). You can generate equations using the application MathType or the built in Equation Editors if you are using MSWord or Pages. It is the most convenient way to work in Word or Pages with mathematical expressions that have many Greek letters and may be useful for your homework answers. Google Docs is not recommended. Assignments are a mix of practical applications using R/Python/Eviews and mathematical proving or derivations.

You need to submit electronically (justin.eloriaga@dlsu.edu.ph) a pdf version of your word-processed assignment. PLEASE NAME the pdf file SURNAME_ECO602M_ProblemSetX. The deadline for electronic submission of the pdf copy is the Friday of the week of submission. Each person is also required to submit the problem set in the AnimoSpace portal for the course.

Empirical Paper Students are to submit a final empirical term paper which concerns any empirical study which is an application of lessons or concepts learned in class. Students are suggested to explore their respective research interests, whether it is in the field of microeconomics or macroeconomics. Other applications may also be explored. Students may opt to make the term paper individually or in pairs.

Students are to submit a concept paper on Week 10 of the term which contains the following

1. Brief Background of the Study
2. Research Objectives
3. Methodology

This concept paper shall be *peer reviewed* using AnimoSpace's peer review function. Give a half page review of what you think the paper you reviewed should improve upon. The professor will likewise review all papers and give a similar review. Please take these reviews into consideration as you build your final project.

At the end of the term, students are to submit a final term paper which contains everything in the concept paper in addition to the following.

1. Brief Review of Related Literature
2. Theoretical Framework
3. Results and Discussion
4. Conclusions

Students are expected to submit just a soft copy of the paper to justin.eloriaga@dlsu.edu.ph in addition to any codes used to generate the results and an excel file of the data used. The paper must be submitted in .pdf form. The deadline will be on the 14th week at a schedule announced by the professor. Each student is also required to submit the .pdf file of the paper on AnimoSpace. The criteria for grading the term paper is seen in the next page. I do not impose a specific format (i.e. Chapter I, Chapter II, Chapter III, etc.). You are all graduate students with the capability to write clearly and in the manner you see fit. I also have no prescribed length. Generally, I think a good term paper can be less than 10 pages in length (excluding references). Do not afraid to be novel in your approach, just do what you think is right. Excellent papers shall be *required* to submit their papers for presentation at the Philippine Economics Summit. Details on that will follow.

Each group shall be required to present their work to the rest of the class during the last meeting of the term. This group presentation shall comprise 15 percent of the total grade allocation for the empirical paper. The presentation should be at most 15 minutes per paper.

Bonus Seatworks Finally, there are two bonus seatworks available. These are not graded but act as bonuses to the problem set. The bonus seatworks are posted on AnimoSpace and are multiple choice or True/False questions. A twenty minute time limit is imposed. Up to a maximum of plus three percent to the final grade is attainable for each seatwork. It is encouraged that students answer the seatwork after they have accomplished the problem sets.

Learning Outcome/Criteria	Excellent (90-100)	Very Satisfactory (80-89)
<i>Technical Sophistication</i>	The review adequately covers all technical aspects needed to carry out the empirical methodology of the paper.	The review satisfactorily covers some technical aspects that are critically needed to carry out the empirical methodology of the paper.
<i>Replicability</i>	The authors did submit and fully documented data and provided log and other relevant files. No problems were encountered in replicating the results.	The authors did submit and fully documented data and provided log and other relevant files but some problems are encountered in replicating the results.
<i>Clarity and degree of testability of hypotheses</i>	The paper's hypotheses are testable and the all steps were taken to achieve the conclusion	The paper's hypotheses are testable but not all steps were taken to achieve the conclusion.
<i>Application of Economic Theories</i>	The paper adequately covers all technical aspects and is able to apply necessary foundations to the full extent.	The paper satisfactorily covers some technical aspects that are critically needed to carry out the empirical methodology of the paper.
Learning Outcome/Criteria	Satisfactory (60-79)	Needs Improvement (0-59)
<i>Technical Sophistication</i>	The review provides a minimal (many of the technical aspects are ignored) yet acceptable coverage of the technical aspects needed to carry out the empirical methodology of the paper.	The review is not helpful and totally irrelevant for carrying out the methodological objectives of the empirical paper.
<i>Replicability</i>	The authors did submit and fully documented data and provided log and other relevant files but there are a lot of problems in replicating the results.	The authors did not submit the dataset or failed to provide log files and other relevant files.
<i>Clarity and degree of testability of hypotheses</i>	The paper's hypotheses are non – testable	The hypotheses are in no way related to the paper's
<i>Application of Economic Theories</i>	The paper provides a minimal (many of the technical aspects are ignored) yet acceptable coverage of the technical aspects needed to carry out the empirical methodology of the paper.	The paper is not helpful and totally irrelevant for carrying out the methodological objectives of the empirical paper.

Learning Resources

The following are the main references of this course

- Brooks, C. (2019). Introductory econometrics for finance. Cambridge university press.
- Eloriaga, J. (2020). Time series econometrics.
- Enders, W. (2008). Applied econometric time series. John Wiley & Sons.
- Hamilton, J. D. (1994). Time series analysis . New Jersey: Princeton.
- Lütkepohl, H. (2013). Introduction to multiple time series analysis. Springer Science & Business Media.

Supplementary references include

- Dreyfus, G. (2005). Neural networks: methodology and applications. Springer Science & Business Media.
- McCandless, G. (2008). The abcs of rbc's. Cambridge, Massachusetts, London: Harvard.
- Wooldridge, J. M. (2016). Introductory econometrics: A modern approach. Nelson Education.

Contact and Consultation Hours

My consultation hours are every Friday from 6:00 pm to 7:00 pm and Saturday from 9:00 am to 11:00 am. Please set an appointment at least one day in advance. Meetings are usually held at the SOE Office Mezzanine Area but if an online Zoom meeting is feasible, that is preferred,

Department Approval

Syllabus prepared by

Justin Raymond S. Eloriaga

Approved by

Dr. Arlene B. Inocencio
Department Chair

Dr. Marites M. Tiongco
Dean

About Your Professor



Justin Raymond S. Eloriaga is a lecturer at the De La Salle University School of Economics and a Data Analytics Specialist assigned to the Bangko Sentral ng Pilipinas. He has taught various courses in both the undergraduate and graduate school with a primary focus on advanced econometrics and microeconomic theory. He obtained his Bachelor of Science in *Applied Economics* from De La Salle University and graduated *summa cum laude*. He also obtained a Master of Science in *Economics* from the same university, completing the rigorous ladderized track in the fastest time recorded in the school's history. He was also the 17th Commissioner of the Young Economist's Convention of the Economics Organization and a founding executive board member of the Lasallian Graduate Economics Society. He was also one of the top young economists of 2019 as inducted by the Philippine Economics Society. He is also a massive fan of Star Wars, as you may have inferred from the lecture notes.

Email: justin.eloriaga@dlsu.edu.ph

Consultation Hours: 18:00 - 19:00 Friday and 9:00 - 11:00 Saturday

Office: *School of Economics Office*. St. La Salle Hall Room 221, Mezzanine Area

