Table of Contents

Module 3: Modeling Heteroskedasticity – Case Studies

3.3: Case Study 1: Modeling Currency Exchange

3.3.1 Exploratory Analysis

In the next three lessons, I will present a case study illustrating the modeling currency exchange time series using the ARMA-GARCH model.

Foreign Exchange Rates

Foreign exchange is the conversion of one country's currency into that of another. From the economic theory point of view, over long periods of time, exchange rates between two countries are established in a way that they will tend to offset the differences in inflation rate between the countries. In other words, in an efficient international economy, exchange rate would give each currency the same purchasing power in its economy. This is called the Purchasing Power Parity. For this analysis, I selected to analyze the exchange rates between the US dollar and three other currencies. The Euro, which is the currency in most countries of the European Union, Brazilian Real, and the Chinese Yuan. The exchange rate between the US dollar and the Euro should best reflect the Purchasing Power Parity Theory. Brazil is a developing country, and hence there may be some fluctuations in terms of the exchange rates due to political instabilities. Last, I selected Chinese Yuan because of recent debate on currency manipulation. All datasets were acquired starting January 1993 until end of September 2020.

Generally, the study of temporal dependencies in exchange rates can be used for forecasting daily fluctuations in rates for investments. Large contracts between countries can be placed given the level of exchange rates. Moreover, investments in a country are also highly dependent on the volatility of the exchange rate with the currency of that country.

Exploratory Data Analysis

Exploratory Data Analysis

```
#Load data: USD/EUR data
    data=read.csv("USD_EUR_Historical_Data.csv",header=TRUE)
    names(data)[1]="Date"
    data$Date=as.POSIXct(as.character(data$Date),format='%d-%b-%y')
    data=xts(data[,2],data[,1])
    colnames(data)="rate"
#Exploratory analysis
    #Plot original exchange rates
        plot(data$rate,type='l',main='USD/EUR Exchange Rate',ylab="Exchange rate")
    #Differencing the series
        diff.rate=diff(data$rate); diff.rate <- diff.rate[!is.na(diff.rate)]
    #Plot differenced series
        plot(diff.rate,type='l',main='USD/EUR Exchange Rate Daily Changes',ylab="Difference")
```

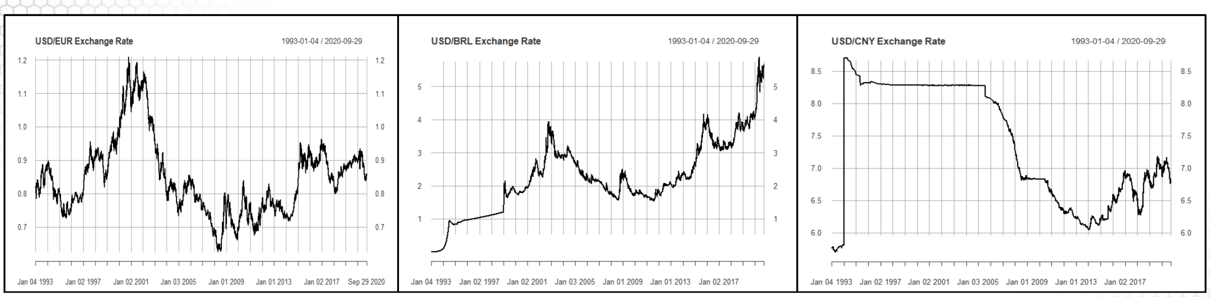
Georgia Tech

Each currency rate is provided in a separate file, but all files have the same structure and hence the R code in this slide, which is for Euro only, applies to other currencies. I will note that I'm using here yet another R command to process dates, which is different than other commands used before. This is to show that there are many R functions to process time series data.

The R code on this slide also provides the R lines of code for plotting the time series data and the one order differenced time series.

Time Series Plots

Time Series Plots



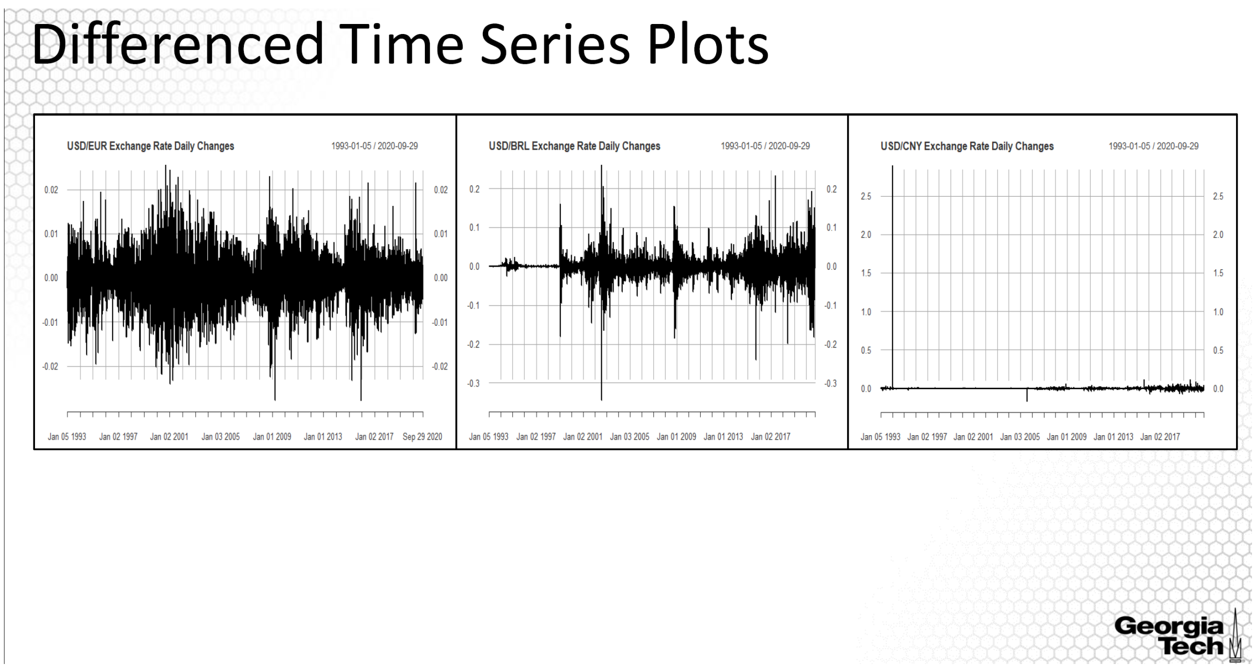
Georgia Tech

The exchange rate between USD and Euro starts at a rate of approximately 0.8, meaning 0.8 Euro for one Dollar, and continues to increase in favor of US dollar reaching a level as high as 1.25 in August 2000, then slowly decreasing over a long period of time with large volatility. Interestingly, at the end of the period of the study, the exchange rate reached a similar level as the rate when the Euro was introduced.

Similarly, the rate between the US dollar and the Brazilian Real had ups and downs varying over a large range, a much larger range than the exchange rate between US dollar and Euro, pointing to higher volatility and hence higher risks in Brazilian investments.

Last, the exchange rate between US dollar and Chinese Yuan has a surprising pattern. For many years, the exchange rate presented little variation and change until January 1994 with a big jump beginning of the year 1994, showing again low variation until 2007, then slowly decreasing until end of 2015 when it started to increase again.

Differenced Time Series Plots

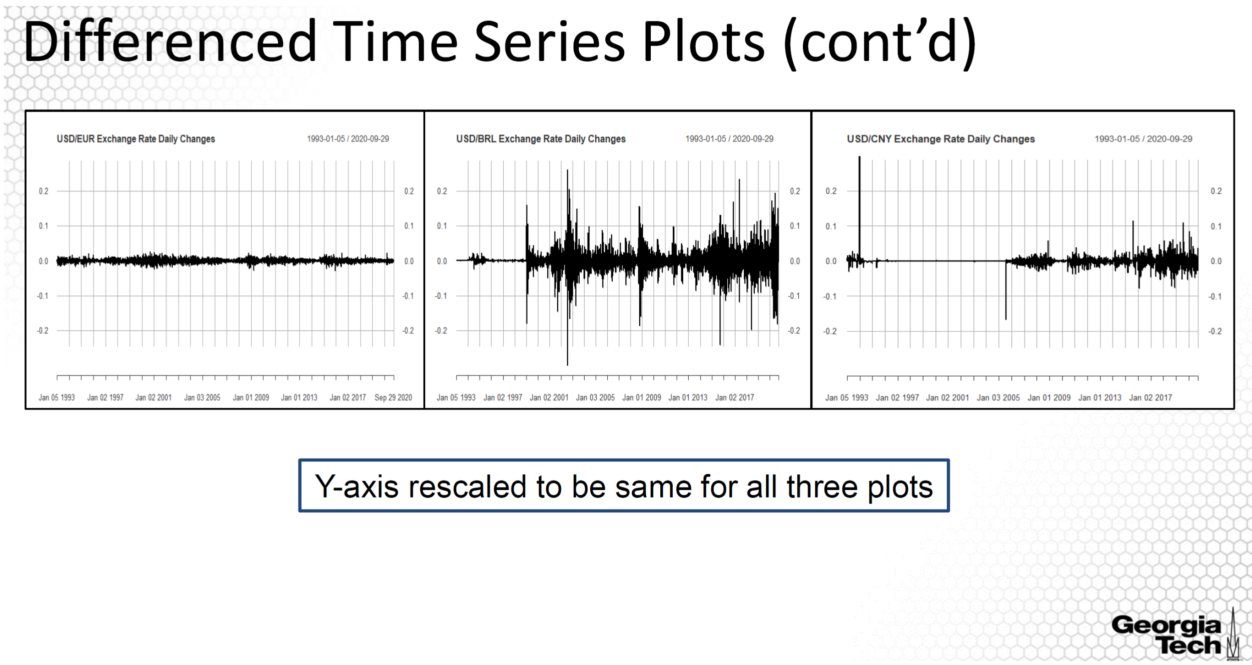


The one-order difference plots are even more interesting since they reveal volatility and irregularities in the exchange rates. The difference time series for the USD-Euro exchange rate points to periods of large volatility followed by periods of small volatility. Interestingly, the volatility in the exchange rate during the covid19 crisis is not necessarily much larger than other periods such as in 2001 or 2009. We also see some sort of cyclical patterns or cluster patterns, with higher volatility increasing say in January 2001 or January 2009 then decreasing over a period of 5-6 years, then increasing again.

The difference plot for the USD-Brazilian Real exchange rate looks very different from the one for the USD-Euro exchange rate. First, the scale is 10 time higher, with periods of very small volatility and very large volatility spread out without a clear pattern. We can also see that the volatility in January 2001 is again much higher than the volatility during the covid19 crisis.

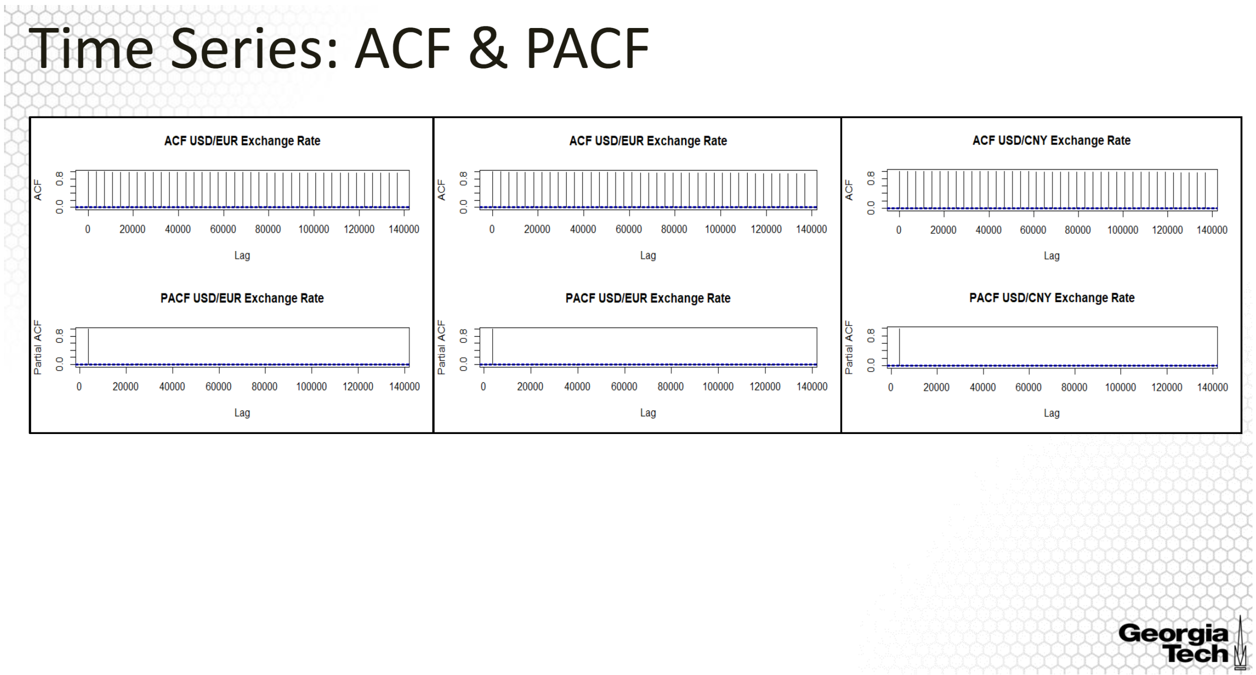
For the exchange currency between US Dollar Gand Chinese Yuan, there is an extreme value in 1994. Because of this outlier or anomaly in the exchange rate, other patterns in the volatility are hard to evaluate.

Differenced Time Series Plots (cont'd)



These plots are for the differenced time series, but re-scaled in a way that all three plots have the same scale. On this scale, the volatility in the USD-Euro exchange rate is much smaller. Once the outlier 1994 for the Chinese Yuan is removed the largest volatility is now for the Brazilian Real. However, for the Chinese Yuan, we identify periods with extreme small changes over long periods of time but also periods of high volatility. Extreme small changes over long periods could be interpreted as control over the currency purchasing power. However, we do note that the volatility in the exchange rate for the Chinese Yuan is larger than for the Euro in recent years.

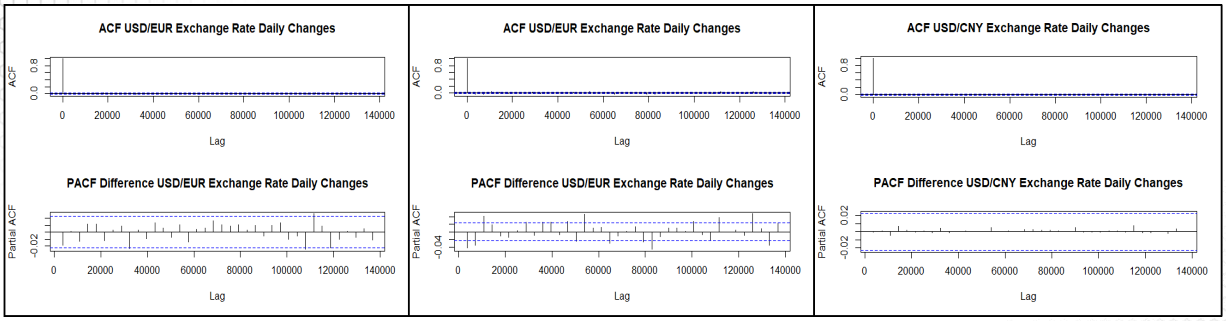
Time Series: ACF & PACF



Next, we will assess the autocorrelation and partial autocorrelation plots for the three time series. The ACF plots are on the first row and the PACF plots are on the second row. The ACF plots point to non-stationarity for all three times series. Because the ACF plots show a slow decay without a pattern, the non-stationarity is primarily due to a temporal trend. The PACF plots only show partial autocorrelation at lag one. Those are also patterns we see in the ACF and PACF plots of a random walk.

Differenced Time Series: ACF & PACF

Differenced Time Series: ACF & PACF



The ACF plots for the differenced times series resemble those of white noise with the sample autocorrelation values being small within the confidence band for lags one and higher. The sample partial autocorrelation is small, within the confidence bands, for all lags. While the difference time series show that they may resemble white noise, we also distinguished time-varying volatility for all three time series. In the next lessons, we will model the three time series using the joint ARMA-GARCH model to assess this assertion. If the selected ARMA model has small orders, it is plausible for the time series to behave like a white noise while the volatility to be varying over time.

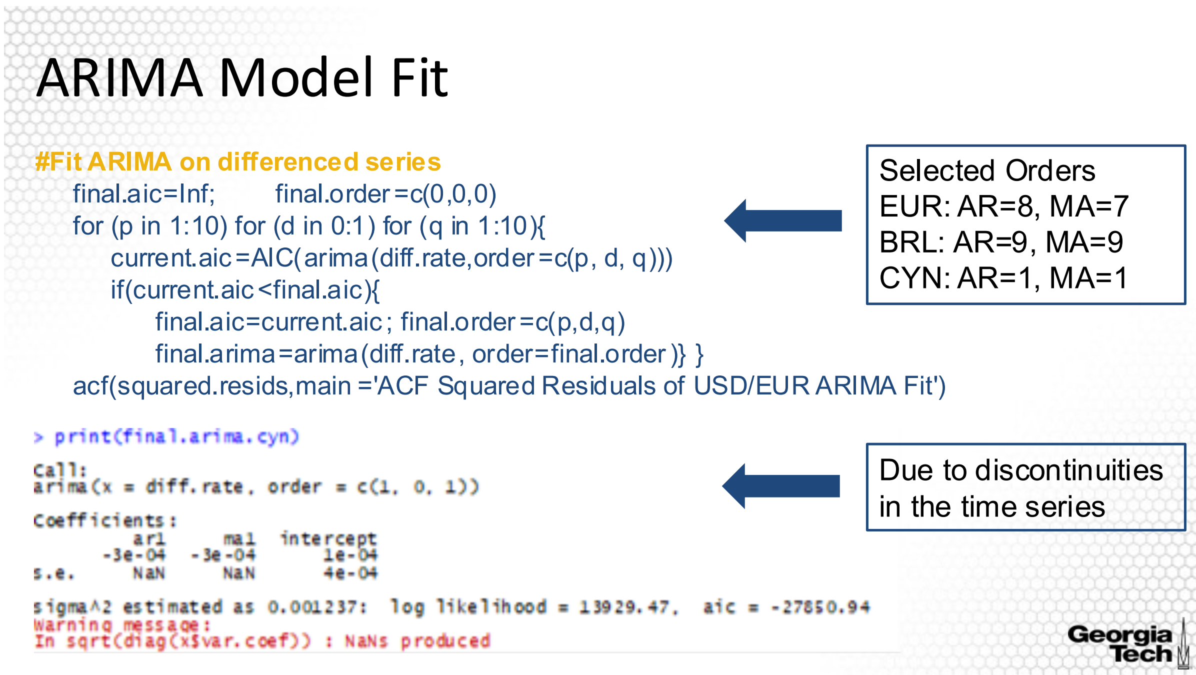
Summary:

In summary in this lesson I introduced an example that will be used to illustrate modeling heteroskedasticity. The example is related to the study of the exchange rate between USD dollar and three other currencies.

3.3.2 ARMA Modeling

In this lesson I will focus on the ARMA-GARCH modeling of the time series of the difference exchange rates.

ARIMA Model Fit



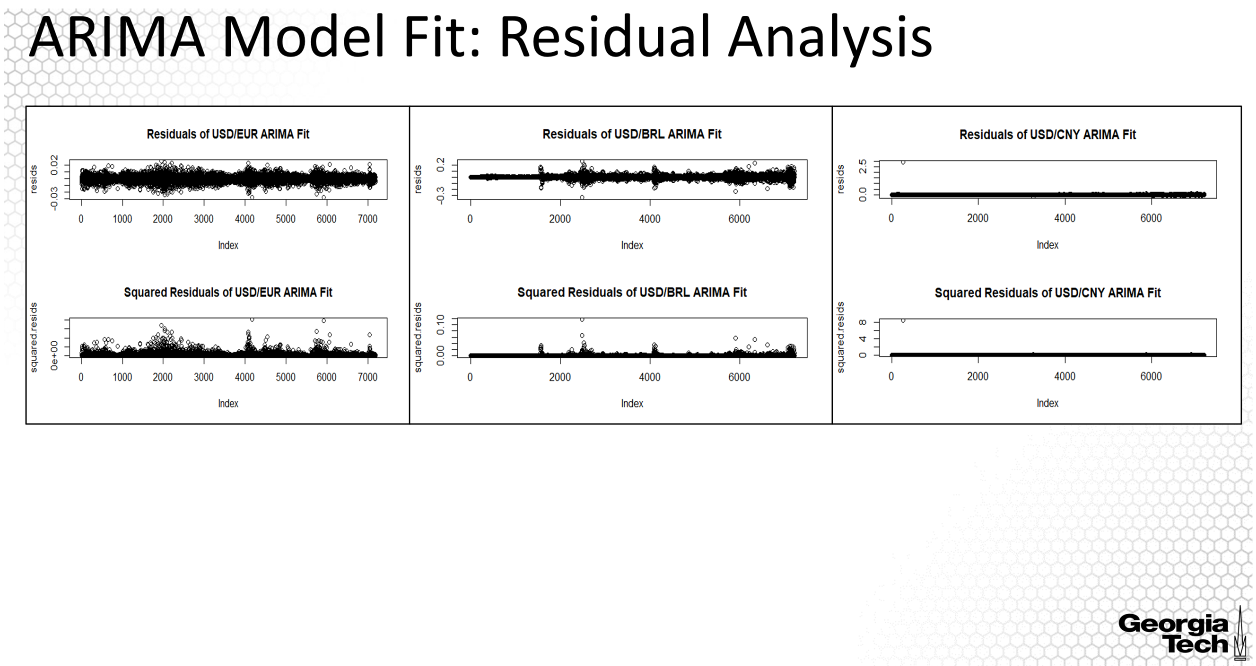
The R code on the slide fits an ARIMA model to the differenced time series of one currency exchange rate, being generalizable to the other two currencies. Note that I applied here the ARMA modeling to the differenced time series instead of the time series itself. For the rest of the analysis, I will apply the ARMA-GARCH modeling to the differenced time series. In this implementation, I allowed for d order of 0 or 1 to evaluate whether another differencing of time series is needed. For the AR and MA orders, I allowed for orders up to 10.

The selected orders are different for the three currencies. Interestingly, the selected orders for the USD-Euro and USD-BRL exchange rate differences are quite large while for USD-CYN are small. A possible explanation of this is that the difference time series corresponding to the first two currencies show clustered patterns in the volatility, which could be instead picked up by the ARMA model for conditional mean. It is expected that when jointly modeling an ARMA-GARCH, the orders of the ARMA to become much smaller.

On the slide find the output from the ARMA model fit for the difference time series of the USD-CYN rate. As noted, the estimated standard errors for the AR and MA coefficients are Nas. As in the warning message shown in red, this is because the estimated variance of the coefficients is not positive. In fact, if we apply the ARMA model to the time series including only the 10-15 years, this issue will not come up.

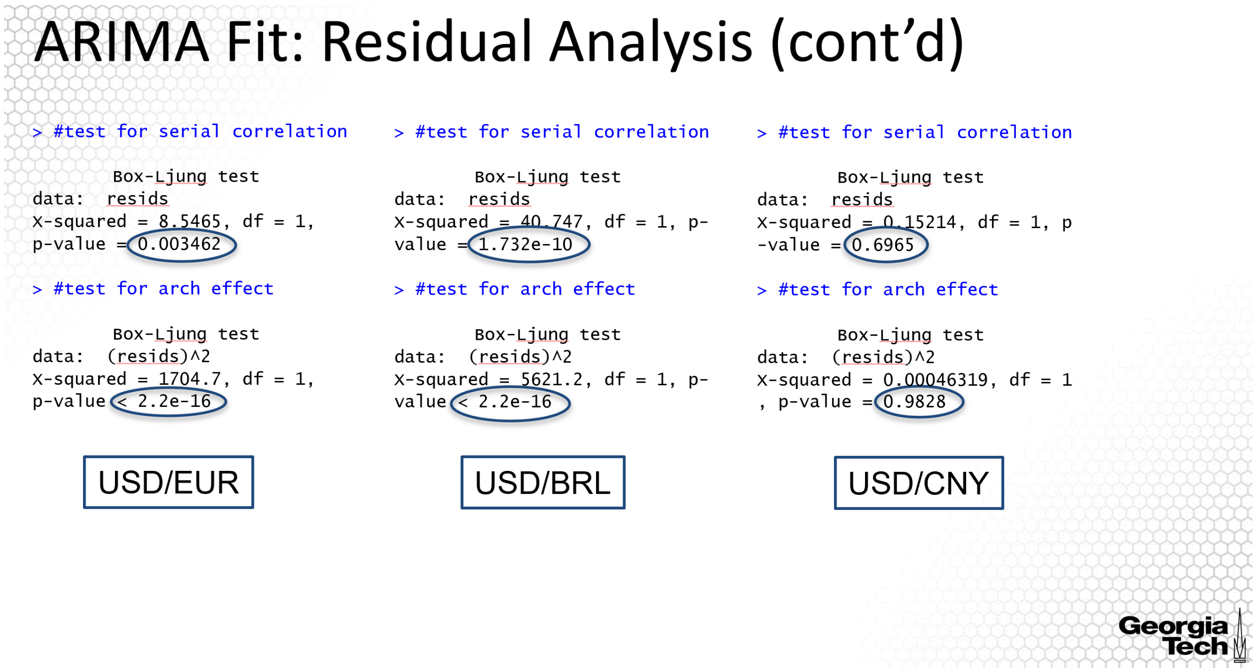
The difference time series for USD-BRL and USD-CYN rates have some discontinuities, which are defined by sudden extreme changes; these are particularly visible for the USD-CYN rate. This in turn, means that the unconditional variance is theoretically not finite hence the estimated variances of the coefficients are not finite. A similar issue we encounter for the USD-BRL rate, but not for the USD-Euro rate.

ARIMA Model Fit: Residual Analysis



Those issues are also revealed in the residual plots. The upper plots are the plots of the residuals, and the lower plots, are of the squared residuals for all three currencies. The residuals for the Euro show some large variability clusters extended over longer periods of time, while the residuals of the Brazilian Real present larger variability clusters over short periods of time. Moreover, the residuals plot for the Chinese yuan presents some extreme values or outliers. Because of these outliers, it is difficult to assess the variability over time. Thus, for these currencies the volatility shows different patterns between long versus short periods of high volatility, and outliers in volatility. Temporal patterns in the squared residuals are difficult to evaluate because of the outliers. Overall, we'll identify nonconstant variance for the residuals of all three currencies.

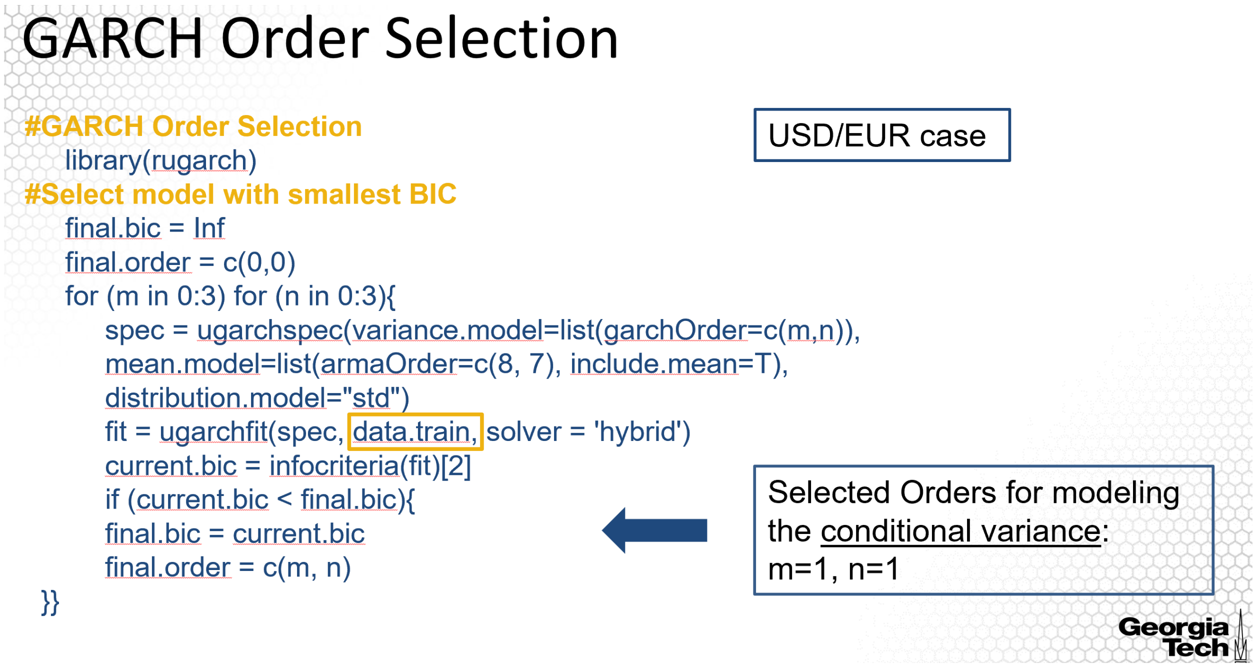
ARIMA Model Fit: Residual Analysis (cont'd)



Last, I applied the testing procedure for uncorrelated data using the box.test R command for serial correlation and for an ARCH effect in the residuals.

The p values for testing for serial correlation in the residuals and squared residuals are large for the Chinese Yuan indicating no serial correlation. But the p-values are small for the Euro and Brazilian real. A potential explanation for not detecting an ARCH effect in the residuals for the Chinese yuan is that the variability in the residuals presents extreme values but otherwise constant over a long period of time. We'll explore the volatility of the three currencies in the next lesson.

GARCH Order Selection



The selected orders for ARMA applied to the USD-Euro exchange rate difference time series were an AR order equal to 8 and an MA order equal to 7. Assuming the ARMA order is fixed for the conditional mean, we now select the order of the GARCH model for the conditional variance, assuming an ARMA-GARCH joint model fit. Note that this time I'm fitting the orders on the training data, which is a subset of the initial time series without the last months of the time series, August and half of September 2020; these last two months will be the testing data to compare the performance of the forecasting the mean and variance. To select the GARCH orders, we loop through all combinations of m and n orders, taking values between 0 and 3, fit the ARMA with orders 8 and 7 and GARCH model with orders m and n; and then select the orders m and n for the model with the smallest BIC. Note that I am using here the BIC to select a parsimonious GARCH model, which is preferable when fitting an ARMA-GARCH joint model, which is already very complex. The selected orders for the GARCH models for the conditional variance are m = 1 and n = 1.

ARMA Order Selection (refined)

ARMA Order Selection (refined)

USD/EUR case

```
#Refine the ARMA order
  final.bic = Inf
  final.order.arma = c(0,0)
  for (p in 0:6) for (q in 0:6){
    spec = ugarchspec(variance.model=list(garchOrder=c(1,1)),
    mean.model=list(armaOrder=c(p, q), include.mean=T),
    distribution.model="std")
    fit = ugarchfit(spec, data.train, solver = 'hybrid')
    current.bic = infocriteria(fit)[2]
    if (current.bic < final.bic){
      final.bic = current.bic
      final.order.arma = c(p, q)
    }
  }
```

Selected Orders for modeling the conditional mean:
AR=2, MA=1

Georgia Tech

Next, we'll continue with refining the orders for ARMA, fixing the orders for GARCH. Here, I'm selecting p and q of the ARMA model for the conditional mean, but this time fitting a joint ARMA-GARCH model. Recall, we select the orders of ARMA and GARCH separately by fine tuning the order selection. Here, we assume that the orders of GARCH are as selected in the previous slide and allow only the ARMA orders to vary within the range 0 to 6.

The selected orders are now AR = 2 and MA = 1. Thus the ARMA orders reduced from 8 to 2 for the AR order and from 7 to 1 for the MA order. As I pointed out earlier, the high ARMA orders are because the ARMA model of the conditional mean is compensating for the time-varying volatility; once the conditional mean and conditional variance are jointly modeled, the ARMA orders decreased significantly.

GARCH Order Selection (refined)

GARCH Order Selection (refined)

USD/EUR case

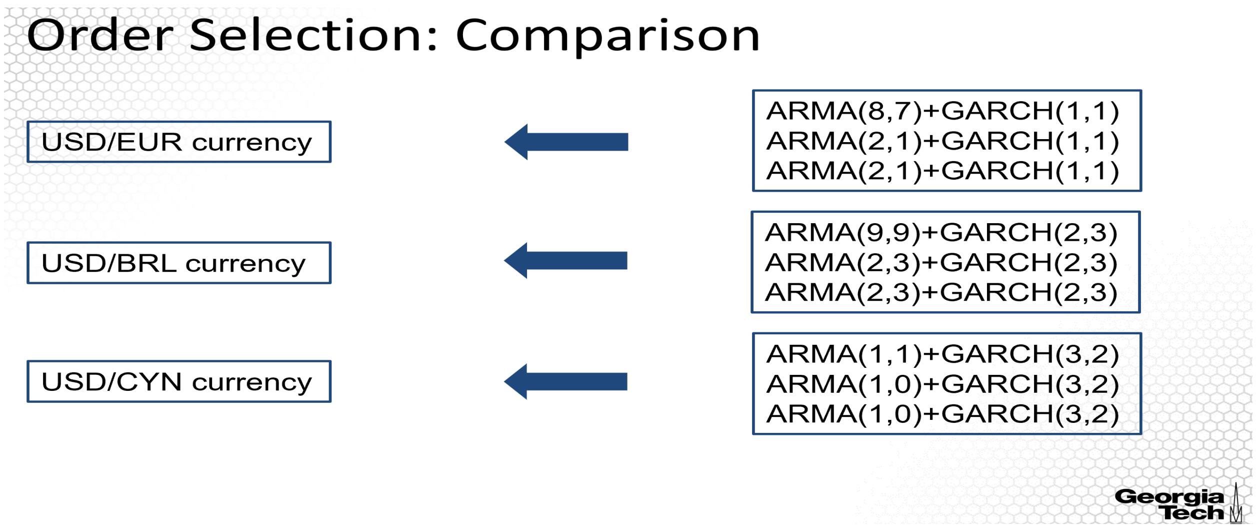
```
#Refine the GARCH order
  final.bic = Inf
  final.order.garch = c(0,0)
  for (m in 0:3) for (n in 0:3){
    spec = ugarchspec(variance.model=list(garchOrder=c(m,n)),
    mean.model=list(armaOrder=c(final.order.arma[1], final.order.arma[2]),
    include.mean=T), distribution.model="std")
    fit = ugarchfit(spec, data.train, solver = 'hybrid')
    current.bic = infocriteria(fit)[2]
    if (current.bic < final.bic){
      final.bic = current.bic
      final.order.garch = c(m, n)
    }
  }
```

Selected Orders for modeling the conditional variance:
m=1, n=1

Georgia Tech

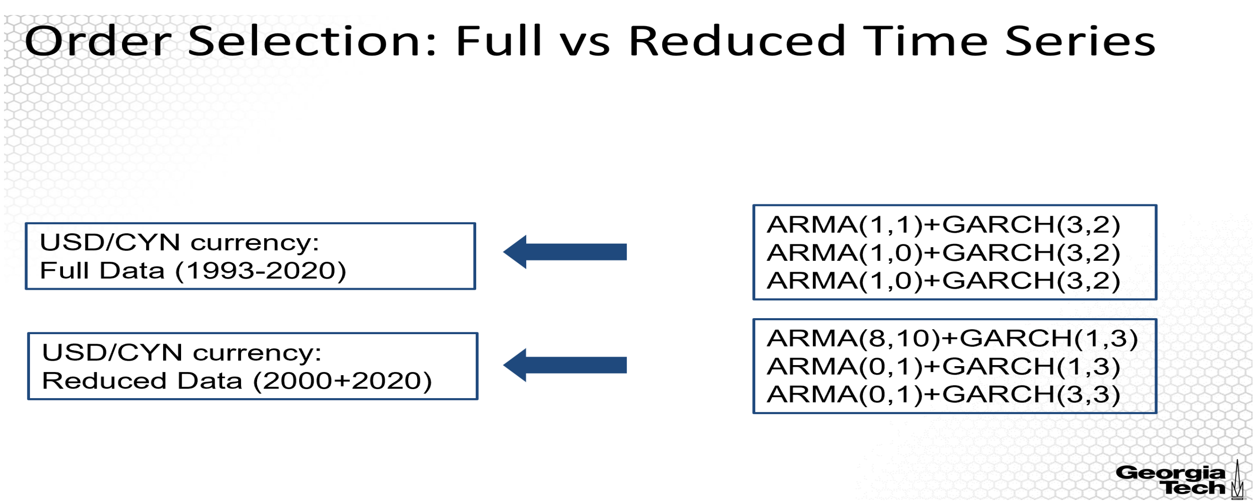
Last, assuming the ARMA order's fixed to the values provided by the model provided on the previous slide, now we fine tune the orders of GARCH, by selecting again among all combinations of m and n taking values between 0 and 3, the orders selected are m = 1 and n = 1, thus with no change from the order selected in the first round of order selection for GARCH.

Order Selection: Comparison



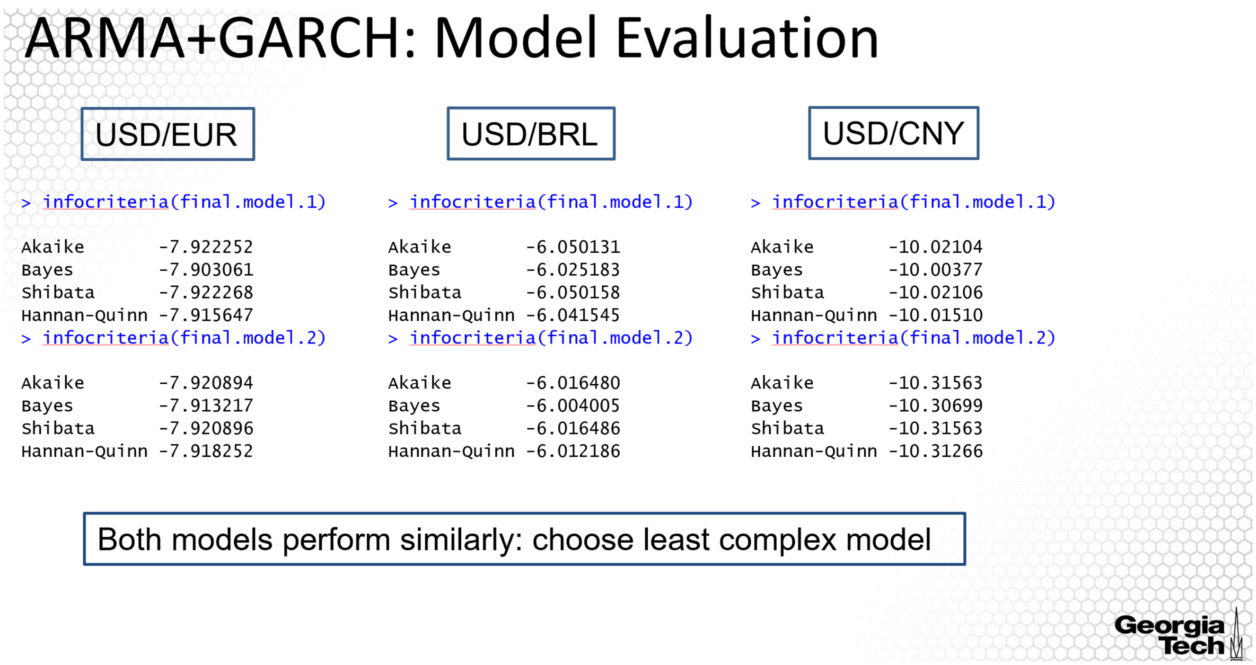
For all three currencies, the selected ARMA after accounting for the joint modeling with GARCH modeling of the conditional variance reduces to a less complex model than initially selected. There are also no improvements in the order selection of GARCH when considering the more complex ARMA versus the less complex ARMA, thus for all three currencies, we'll only compare two models, the first one with the complex ARMA and the third with the less complex ARMA.

Order Selection: Full vs Reduced Time Series



As I pointed out earlier, the USD-CYN exchange rate has discontinuities before 2000. For that, I applied the ARMA-GARCH model selection to the reduced time series including only data for years 2000 to 2020. Here I am comparing the order selection for both the full and reduced time series. For the reduced time series, the order selection for the ARMA is 8,10, much larger than the selected order for the full time series. This result is similar to the order selection for the other two currencies. However, the final model for the reduced time series is similar to that for the full time series. In the next lesson, we will compare how the forecasts will change when comparing the ARMA-GARCH models for the full and reduced time series.

ARMA+GARCH: Model Evaluation



Next, I compare the two models for each currency exchange rate using multiple information criteria. On the slide, there are the outputs for the comparison of the information criteria.

Consistently across all three time series, both models perform similarly, suggesting that the less complex model with low AR and MA orders for the conditional mean would perform similarly as the more complex ARMA with higher AR and MA orders, hence being the preferred model.

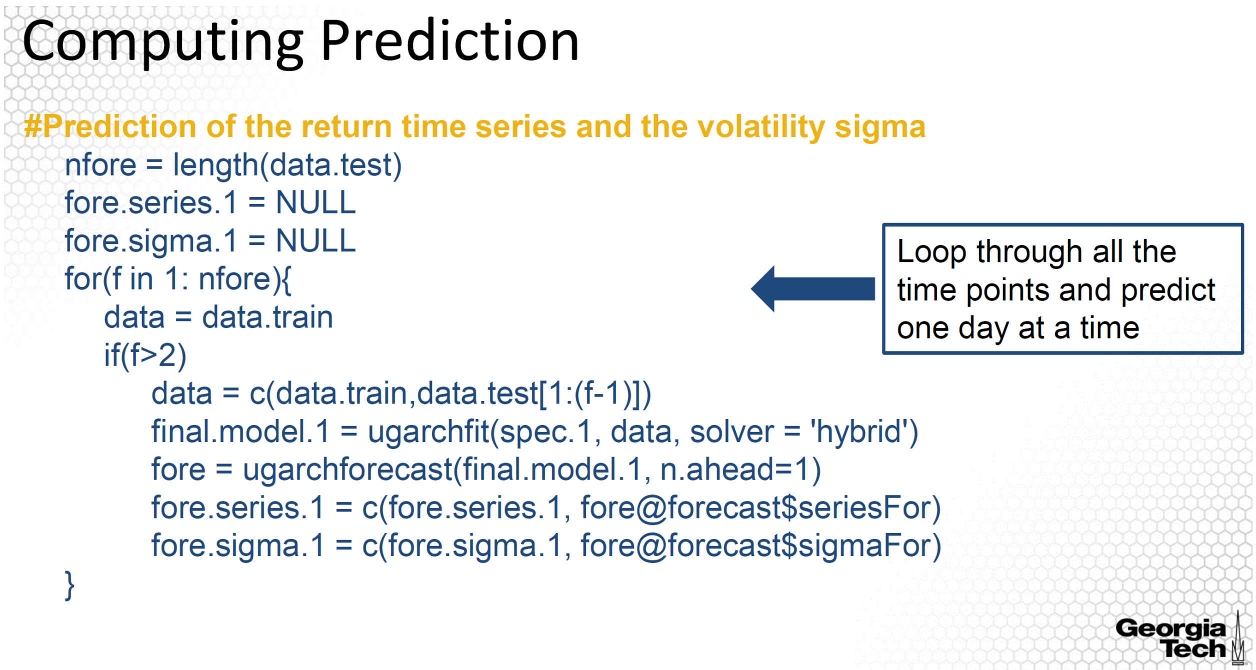
Summary:

To summarize, in this lesson I performed the ARMA-GARCH modeling of the difference time series for the USD-Euro, USD-BRL and USD-CYN currency rates.

3.3.3 Prediction

In this lesson I will continue the analysis of the three currency rates with a focus on the prediction.

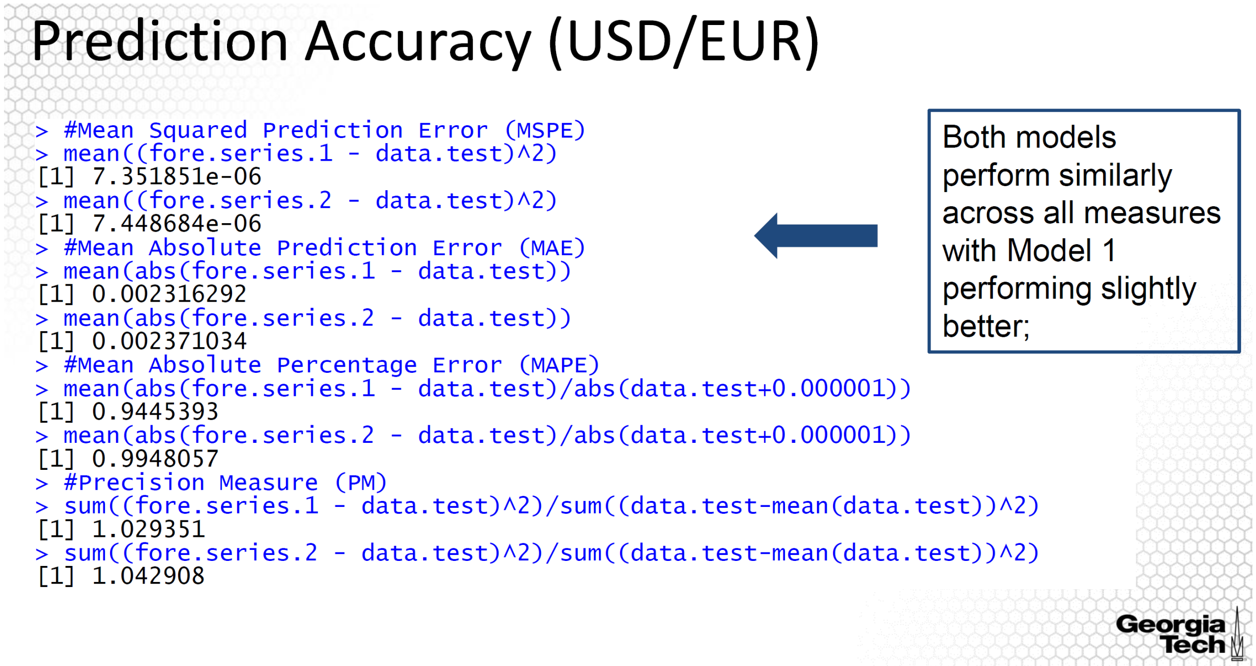
Computing Prediction



For this example, we predict the currency for the last month and a half of the time period, the months of August and half of month of September 2020, representing the testing data. The predictions are one lag ahead obtained on a rolling basis.

That is, we loop through all the time points to be predicted and predict only 1 day ahead. Specifically, we set the past data as being the time series observed up to the time point at which a prediction will be made, then apply the ARMA-GARCH model to the time series observed up to the time point at which a prediction will be made. Thus, with each prediction the training data set changes. After the model fit, the one lag prediction for both the mean and variance are obtained, using the ugarchforecast R command. We thus obtain predictions at each time point within the time period of the testing data. We compare the predictions for both models considered for each currency as introduced in the previous lesson, where the difference in the two models is in the specification of the ARMA orders, the first model being more complex than the second. The code on the slide is only for model 1.

Prediction Accuracy (USD/EUR)



The prediction accuracy measures for the forecast of the mean for the differenced USD-Euro exchange rate are on this slide. The measures are computed for both models.

The performance of the two models is similar with the prediction accuracy being slightly lower for the first model across all measures. As pointed out in the previous lessons, generally the precision measure is most appropriate in comparing prediction accuracy within this type of modeling. The smaller this measure is, the more accurate a prediction is. The measure is approximately 1.04 indicating that the proportion between the variability of the prediction versus the variability of the testing data is 1.04. Thus, the variability prediction is similar to that in the data.

Prediction Accuracy (USD/BRL)

Prediction Accuracy (USD/BRL)

```
> #Mean Squared Prediction Error (MSPE)
> mean((fore.series.1 - data.test)^2)
[1] 0.005115842
> mean((fore.series.2 - data.test)^2)
[1] 0.005165258
> #Mean Absolute Prediction Error (MAE)
> mean(abs(fore.series.1 - data.test))
[1] 0.05764268
> mean(abs(fore.series.2 - data.test))
[1] 0.05852945
> #Mean Absolute Percentage Error (MAPE)
> mean(abs(fore.series.1 - data.test)/abs(data.test+0.000001))
[1] 1.458612
> mean(abs(fore.series.2 - data.test)/abs(data.test+0.000001))
[1] 1.418509
> #Precision Measure (PM)
> sum((fore.series.1 - data.test)^2)/sum((data.test-mean(data.test))^2)
[1] 1.011352
> sum((fore.series.2 - data.test)^2)/sum((data.test-mean(data.test))^2)
[1] 1.021121
```

Both models perform similarly across all measures with Model 1 performing slightly better

Georgia Tech

These are the prediction accuracy measures for the mean prediction for the Brazilian Real currency exchange. Similar results are noted as for the Euro currency. For all measures except the precision measure, the prediction accuracy is slightly worst for the Brazilian Real currency than for Euro. This is not the case for the precision measure, which is smaller for the Brazilian Real currency.

Prediction Accuracy (USD/CYN): 1993-2020

Prediction Accuracy (USD/CYN): 1993-2020

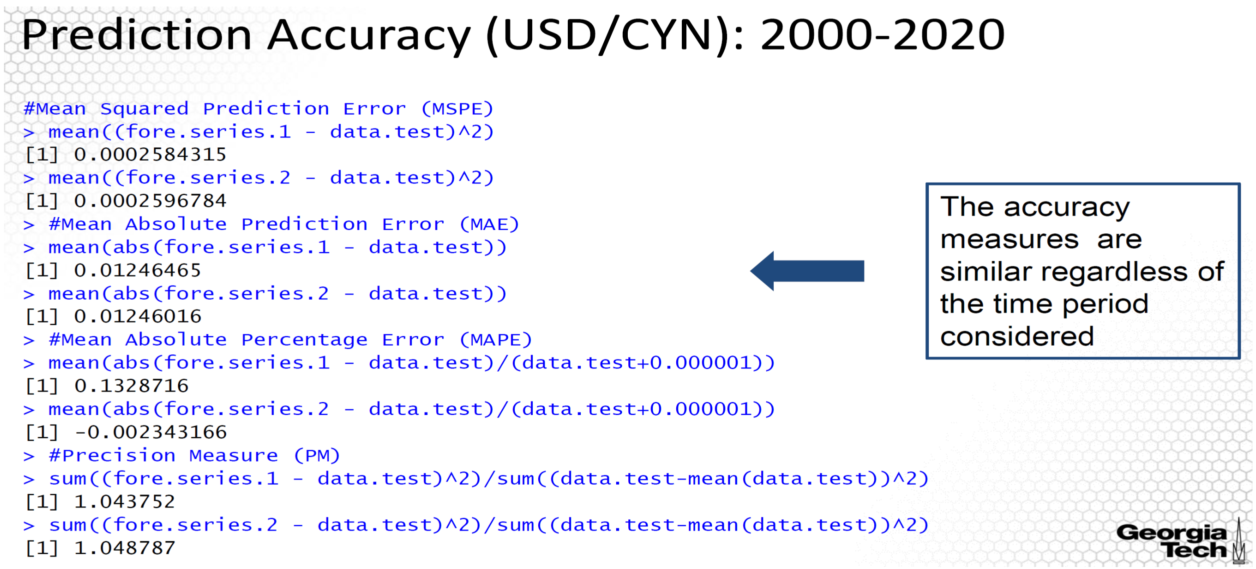
```
> mean((fore.series.1 - data.test)^2)
[1] 0.000260578
> mean((fore.series.2 - data.test)^2)
[1] 0.0002593786
> #Mean Absolute Prediction Error (MAE)
> mean(abs(fore.series.1 - data.test))
[1] 0.01286297
> mean(abs(fore.series.2 - data.test))
[1] 0.012644
> #Mean Absolute Percentage Error (MAPE)
> mean(abs(fore.series.1 - data.test)/abs(data.test+0.000001))
[1] 1.396633
> mean(abs(fore.series.2 - data.test)/abs(data.test+0.000001))
[1] 0.994802
> #Precision Measure (PM)
> sum((fore.series.1 - data.test)^2)/sum((data.test-mean(data.test))^2)
[1] 1.052421
> sum((fore.series.2 - data.test)^2)/sum((data.test-mean(data.test))^2)
[1] 1.047577
```

Both models perform similarly across all measures with Model 2 performing slightly better

Georgia Tech

These are the prediction accuracy measures for the mean prediction for the Chinese Yuan USD currency including the data from 1993 until 2020. The performance of the two models is similar with the prediction accuracy being slightly lower for the second model across all measures.

Prediction Accuracy (USD/CYN): 2000-2020

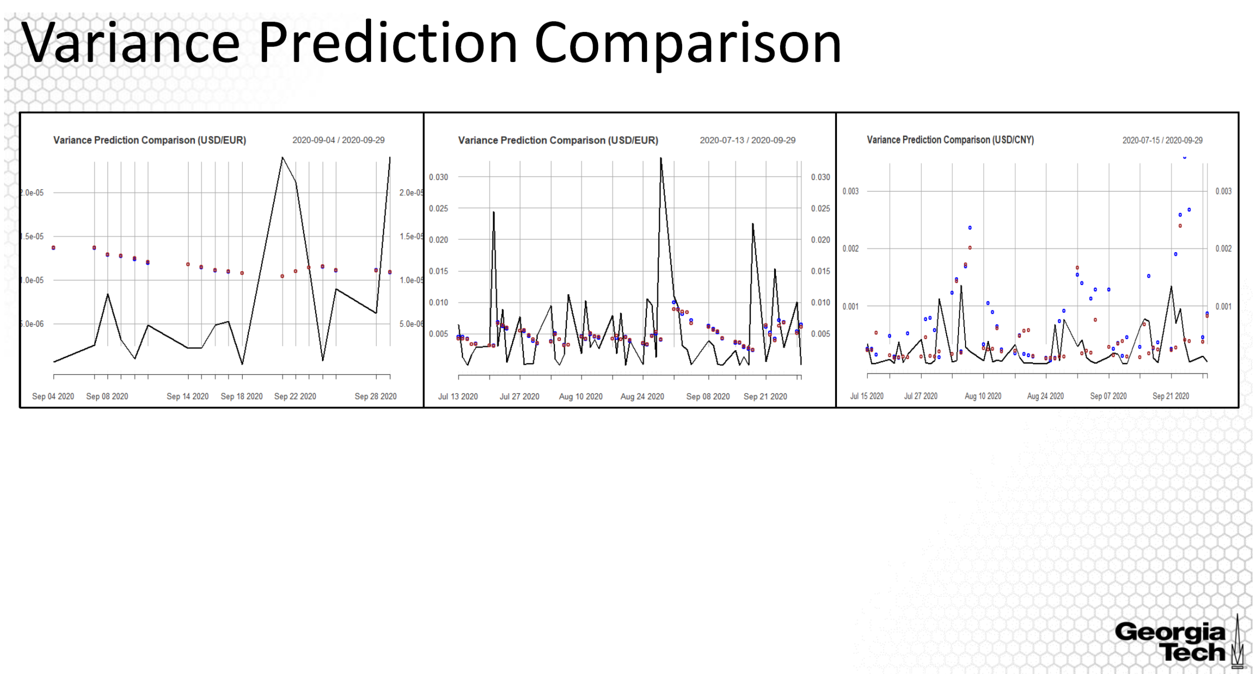


Now I am considering the predictions for the reduced time series from 2000 to 2020, not including periods of discontinuities and currency control. The accuracy measures are similar regardless of the time period considered. This is because the forecasts for the conditional mean are highly influenced by the most recent time series data.

Mean Prediction Comparison

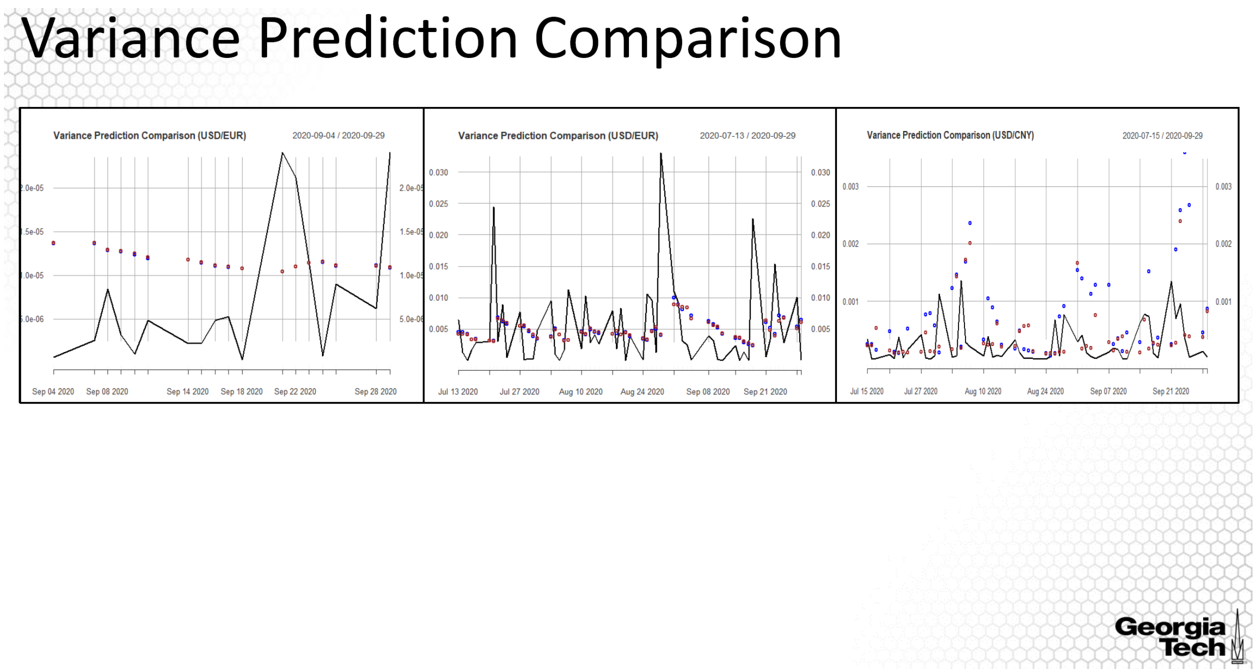
On the slide are the plots comparing the prediction of the conditional mean based on the two models for each of the three currencies. The predicted means are close to zero, with some variations in the predictions derived using the more complex ARMA-GARCH model for the Brazilian Real and Chinese Yuan currencies. Note that the predictions for the conditional mean do not capture the variations in different time series since these variations are due to the conditional variance, not to the mean of the process.

Variance Prediction Comparison



Last, we'll take a closer look at the volatility forecasts. The comparison now is between the squared time series data and the predictions of the conditional variance or volatility. The plots for the three currencies are on the slide. The volatility is not well captured by the forecasted conditional variance for the Euro currency, but it is captured much better for the other two currencies, particularly, much better for the Chinese Yuan.

Variance Prediction Comparison: USD/CYN



I am also comparing here forecasts for the volatility of the USD-CYN currency rate using the full and reduced time series data. Model 1 using the full data and Model 2 using the reduced data capture better the time variations in the volatility with the Model 2 using the reduced data showing over-prediction. Similarly to the assessment of the prediction accuracy of the forecasts for the conditional mean, using the reduced data doesn’t provide a considerable advantage. Again this is because the forecasts are highly influenced by most recent data.

Other Models: R Implementation

Other Models: R Implementation

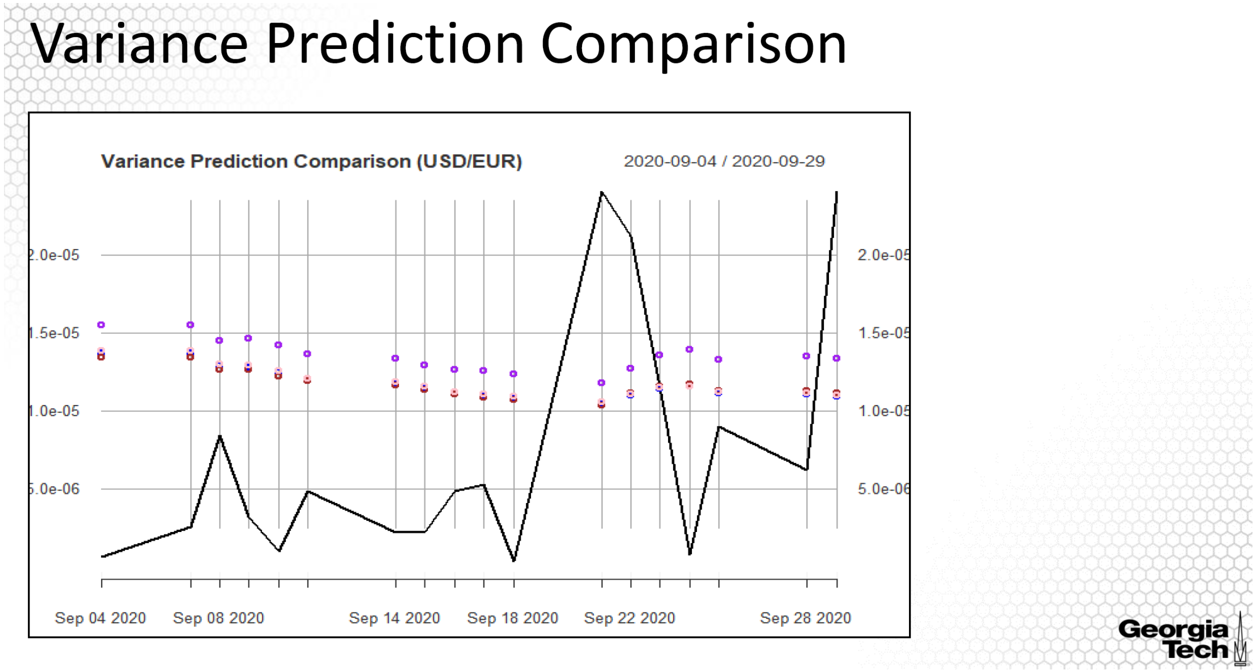
```
#GARCH
spec.1 = ugarchspec(variance.model=list(garchOrder=c(1,1)), mean.model=
      list(armaOrder=c(2,1), include.mean=T), distribution.model="std")
#GJR-GARCH
spec.2 = ugarchspec(variance.model=list(model ="gjrGARCH",garchOrder=c(1,1)),
      mean.model=list(armaOrder=c(2,1), include.mean=T), distribution.model="std")
#EGARCH
spec.3 = ugarchspec(variance.model=list(model ="eGARCH",garchOrder=c(1,1)),
      mean.model=list(armaOrder=c(2,1),include.mean=T), distribution.model="std")
#APARCH
spec.4 = ugarchspec(variance.model=list(model ="apARCH",garchOrder=c(1,1)),
      mean.model=list(armaOrder=c(2,1), include.mean=T), distribution.model="std")
#IGARCH
spec.5 = ugarchspec(variance.model=list(model ="iGARCH",garchOrder=c(1,1)),
      mean.model=list(armaOrder=c(2,1), include.mean=T), distribution.model="std")
```

Georgia Tech

The GARCH model applied so far provides the basis for the development of many other extensions of heteroskedasticity models. In a previous lesson in this module, I discussed several other extensions such as exponential GARCH or integrated GARCH. These extensions and others provided on this slide are meant to deal with different distributions of the volatility. For example, asymmetrical response to positive and negative shocks, large or small kurtosis, meaning more or less heavy tailed distributions among others, or non-linear relationships with the past conditional variance. The good news is that many of the models are already implemented in the R statistical software and most of them can be applied through the commands in the rugarch library used to fit the joint ARMA-GARCH models.

In order to specify the other model types, you will need to specify the model in the model option of ugarchspec(). The default is the GARCH model. Here are four different models.

Variance Prediction Comparison



Let's compare these models using the difference time series of USD-Euro exchange rate. Because we have learned from the joint modeling of the conditional mean and conditional variance, that the conditional mean does not vary with time, I focus here on the modeling of the conditional variance because the different modeling approaches are for the GARCH portion of the model. The code for forecasting the conditional variance is similar as for the GARCH model. On the slide is the plot of the volatility forecast, along with the observed squared time series shown in black. We see that the other models have not improved the estimation of the volatility since none of the predictions captured the ups and downs in the squared difference time series. It's possible that such variations are due to randomness alone rather than temporal variations in the conditional variance. I'll note here that running the forecasting code with all the five models and for a large number of time points at which we obtain predictions took more than a day of computation and time. Such models are computationally expensive. Hence, real time predictions are not possible except when using super computers, which are quite common in the financial industry!

Take Home Conclusions

Take Home Conclusions

- Exchange currencies behave differently primarily with respect to the volatility in the change of the exchange rate.
- Volatility in the USD-Euro is much lower than that for USD-BLR and USD-CYN.
- Predicted volatility captures the temporal variations better for the exchange rate changes with large volatility.
- Different time periods of the same time series can result in different predictions although not necessarily of better or worse accuracy as illustrated with the USD-CYN exchange rate.

Georgia Tech

Overall, we have seen that a joint ARMA-GARCH model performs differently across different time series. Exchange currencies behave differently with respect to the volatility in the change of the exchange rate. Volatility in the USD-Euro is much lower than that for USD-BLR and USD-CYN. Predicted volatility captures the temporal variations better for the exchange rate changes with large volatility. Different time periods of the same time series can result in different predictions although not necessarily of better or worse accuracy as illustrated with the USD-CYN exchange rate.

Summary:

This lesson concludes the data analysis of the exchange rates of the three currencies vs USD. This data analysis also concludes Module 3 of this course.