http://denizstij.blogspot.com/2013/11/cointegration-tests-adf-and-johansen.html

Running VAR model for Multivariate time series.

Multivariate time series analysis is used when one wants to model and explain the interactions and comovements among a group of time series variables

- http://faculty.washington.edu/ezivot/econ584/notes/multivariatetimeseries.pdf

Granger Causality One of the main uses of VAR models is forecasting.

The following intuitive notion of a variable's forecasting ability is due to Granger (1969).

• If a variable, or group of variables, y1 (social media mentions) is found to be helpful for predicting another variable (sales volume), or group of variables, y2 then y1 is said to Granger-cause y2; otherwise it is said to fail to Granger-cause y2.

VAR Model Building and Evaluation steps:

1. Split Raw Clem sales and social data into train and validation (1 year).
2. Based on the # of observation split the ARIMA values into train and validation (1 year).
3. Run the VAR model on training set and forecast sales, social and measure the prediction accuracy by comparing the validation set.
4. Make plots of sales, social and arima (benchmark)
5. Run the model on validation set.
6. Make plots of sales, social and arima (benchmark)
7. Calculate the difference / lift / between sales arima forecasts and sales forecasts from var model.

```
library(vars)
library(astsa)

# Read Clementine Google Trends data in for exogenous variable in VAR model
setwd("/Users/kevalshah/Keval_Backup/University/UChicago/Capstone/Data/Data Clean up/Clean data to be used
clem_google_trends <- read.csv("Clem_Google_Trends_Searches.csv")

# Plot sales, social media and google trends

x1 <- clem_data_sales_ts$Date
y1 <- clem_data_sales_ts$salestszSeasonallyAdjusted
y2 <- clem_data_social_ts$Total.social.media
y3 <- clem_google_trends$clementine.Searches

#plot( x1, y1, type="l", col="red", main = "Clementines Sales, Social #Media and Google Trends", xlab = "Da
#par(new=TRUE)
#plot( x1, y2, type="l", col="blue", ylab = "Social", lwd = "2.5")
#par(new=TRUE)
#plot( x1, y3, type="l", col="orange", ylab = "Social & GT", lwd = "2.5")
#legend("top", legend=c("Sales", "Social"), col=c("red", "blue"), lwd = #2.5, cex=0.8)


# Run VAR Model on Training set

length(clem_data_social_ts$Total.social.media)
length(clem_data_sales_ts$salestszSeasonallyAdjusted)
length(clem_data_sales_ts$Date)
length(clem_google_trends$clementine.Searches)

Train_clem_sales <- clem_data_sales_ts[1:208,2]
Train_clem_week <- clem_data_sales_ts[1:208,1]
Train_clem_social <- clem_data_social_ts[1:208,8]
Train_clem_google_trends <- clem_google_trends[1:208,3]
```

```
# Endogenous variables
Train_VAR_clem <- cbind(Train_clem_sales, Train_clem_social)

#VAR Select
#VARselect(Train_VAR_clem, lag.max = 10, type = "both", exogen = cbind(x3 #=Train_clem_google_trends))

VARselect(Train_VAR_clem, lag.max = 10, type = "both")


Train_VAR_model_clem <- VAR(Train_VAR_clem, p=9, type="both")
summary(Train_VAR_model_clem)
```

The adjusted R-Squared of 87% for equation predicting sales as dependent variable and endogenous variables of social media and sales with lag order of 2 and exogenous variable of google trends with constant and trend deterministic variable indicates a good fit. On the other hand, the inverse, of predicting social media with sales as predictors has Adj. R-Squared of 67%.

Now, we fit our training model on our validation set and check the prediction error / accuracy.

```
# Run VAR Model on Validation / Test Set

Test_clem_sales <- clem_data_sales_ts[209:260,2]
Test_clem_week <- clem_data_sales_ts[209:260,1]
Test_clem_social <- clem_data_social_ts[209:260,8]
Test_clem_google_trends <- clem_google_trends[209:260,3]

length(Test_clem_sales)
length(Test_clem_week)
length(Test_clem_social)
length(Test_clem_google_trends)

# Endogenous variables
#Test_VAR_clem <- cbind(Test_clem_sales, Test_clem_social)

#VAR Select
#VARselect(Test_VAR_clem, lag.max = 10, type = "both", exogen = cbind(x3 =Test_clem_google_trends))

#Test_VAR_model_clem <- VAR(Test_VAR_clem, p=7, type="both", exogen = cbind(x3 =Test_clem_google_trends))
#summary(Test_VAR_model_clem)



# Clementine prediction

#var_train_forecasts <- predict(Train_VAR_model_clem, n.ahead = 52, ci = 0.95, dumvar = #cbind(x3 =Test_cle

var_train_forecasts <- predict(Train_VAR_model_clem, n.ahead = 52, ci = 0.95)



summary(var_train_forecasts)
head(var_train_forecasts)

plot(var_train_forecasts, type = "l", main = "Clem sales + social forecast using train model on test set")

# Check accuracy of our forecasts using train model on test data

# Clem sales volume forecast
accuracy(var_train_forecasts$fcst$Train_clem_sales[,1], Test_clem_sales)

# Clem social media forecast
accuracy(var_train_forecasts$fcst$Train_clem_social[,1], Test_clem_social)
```

Plot Raw Sales and Social media with Forecasts

```
# Append raw + forecast
clem.sales.VAR.All <- append(Train_clem_sales, var_train_forecasts$fcst$Train_clem_sales[,1])
clem.social.VAR.All <- append(Train_clem_social, var_train_forecasts$fcst$Train_clem_social[,1])
clem.week.All <- append(Train_clem_week, Test_clem_week)

clem_df_total <- data.frame(clem.week.All, clem.sales.VAR.All, clem.social.VAR.All)


mar.default <- c(3,3,3,3) + 0.1
par(mar = mar.default + c(0, 1, 0, 0))
plot(clem_df_total[,1:2], type="l",
     ylab="Sales Volume", xlab="Time (Year)",
     lwd=3, main="Clementine: VAR Model", col="hotpink")
par(new=TRUE)
plot(clem_df_total[,3], type="l", col="darkolivegreen1", axes=FALSE,
     ylab="", xlab="", lwd=3)
axis(4)
mtext("Social Media Mentions", side=4, line=+2, adj=0.5)
abline(v=208, lty=3, lwd=3)
legend("top", legend=c("Sales Volume", "Social Media Mentions"),
       col=c("hotpink","darkolivegreen1"), lwd=3, cex=0.75)
```

Comparing sales prediction to ARIMA benchmark, make plots and calculate lift

```
# Calculate accuracy of arima sales forecast and VAR model sales forecast to actual sales values in
# test set
# Compare prediction error of each and calculate the lift obtained from prediction error.


# Difference in Predictions Clementines

All_Clem_Forecasts <- cbind.data.frame(Test_week = as.Date(Test_clem_week), AR=forecast.clem.sales.arima$me
                  ACTUAL=Test_clem_sales)
head(All_Clem_Forecasts)

# Calculate the RMSE. Predicted - Actual values.
AR.error <- forecast.clem.sales.arima$mean - Test_clem_sales
ar.clem.sales.rmse <- sqrt(mean(AR.error^2))
# Calculate MAE
ar.clem.sales.mae <- mean(abs(AR.error))


# Calculate the RMSE. Predicted - Actual values.
VAR.error <- var_train_forecasts$fcst$Train_clem_sales[,1] - Test_clem_sales
var.clem.sales.rmse <- sqrt(mean(VAR.error^2))
# Calculate MAE
var.clem.sales.mae <- mean(abs(VAR.error))


# Calculate Lift in prediction accuracy
paste(round((((ar.clem.sales.rmse - var.clem.sales.rmse)/ar.clem.sales.rmse))*100, digits = 2), "%", sep =
paste(round((((ar.clem.sales.mae - var.clem.sales.mae)/ar.clem.sales.mae))*100, digits = 2), "%", sep = "")

# Create a table to compare RMSE and MAE
accuracy_table <- matrix(c(1082234,790863.8,"26.92%",934652.2,672441,"28.05%"),ncol=3,byrow=TRUE)
colnames(accuracy_table) <- c("ARIMA","VAR", "Lift in Prediction accuracy")
rownames(accuracy_table) <- c("RMSE","MAE")
accuracy_table

# Append ARIMA sales data and forecasts
clem.sales.arima.All <- append(clem_train_sales_data_arima, forecast.clem.sales.arima$mean)

clem.sales.actual.All <- append(Train_clem_sales, Test_clem_sales)

# Create a dataframe w Raw sales data, VAR Forecasts, ARIMA Forecasts and
# Test data.
```

```
clem_df_total_final <- cbind.data.frame(clem.week.All, clem.sales.VAR.All, clem.sales.actual.All, clem.sale


mar.default <- c(3,3,3,3) + 0.1
par(mar = mar.default + c(0, 1, 0, 0))
plot(clem_df_total_final[,1:2], type="l",
     ylab="", xlab="Time (Year)",
     lwd=3, main="Clementine Sales Volume Forecasts Comparison", col="hotpink")
par(new=TRUE)
# Actual data train + test
plot(clem_df_total_final[,3], type="l", col="darkolivegreen1", axes=FALSE,
     ylab="", xlab="", lwd=3)
par(new=TRUE)
# ARIMA Forecast
plot(clem_df_total_final[,4], type="l", col="blue", axes=FALSE,
     ylab="", xlab="", lwd=3)

abline(v=208, lty=3, lwd=3)
legend("topleft", legend=c("VAR", "Actual", "ARIMA"),
       col=c("hotpink","darkolivegreen1","blue"), lwd=3, cex=0.75)
```

Based on the above plot we can see that VAR model which includes social media and lagged sales volume as endogenours predictors and google trends as exogenous variables does predict better than ARIMA model series predicting itself.

```
print(accuracy_table)
```

Based on Root mean squared error and Mean Absolute Error which calculates the prediction error (predicted - actual), from our train and test split, we see increased accuracy of more than 25% when using VAR models.

Predictions for 2015

```
var_clem_forecasts_2015 <- predict(Train_VAR_model_clem, n.ahead = 104, ci = 0.95)

plot(var_clem_forecasts_2015, type = "l", main = "2015 Cupcakes sales + social forecast using train model c
```