Vector Autoregression Model Performance

Gwen Rino

May 23, 2018

Model Description

Vector Autoregression allows for multivariate time series modeling, so in this model I made use of all five variables identified by `randomForest::importance()` and `earth::evimp()` (see "dengue/src/FeatureSelection/FeatureSelection.R"). These variables are:

- nonres_guests
- station_max_temp_c
- reanalysis_tdtr_k
- reanalysis_dew_point_temp_k
- reanalysis_specific_humidity_g_per_kg

Experimentation with the parameter p (number of lags in the autoregression) showed that a model with p=1 resulted in residuals = white noise (see "dengue/src/Models/VAR.R").

```
# Fit model
fitvar1 <- VAR(ts.selected, p=1, type = "both")
fitvar1$varresult$total_cases
```

```
##
## Call:
## lm(formula = y ~ -1 + ., data = datamat)
##
## Coefficients:
##                            total_cases.l1
##                                 9.555e-01
##                          nonres_guests.l1
##                                -2.707e-05
##                     station_max_temp_c.l1
##                                 7.801e-04
##                      reanalysis_tdtr_k.l1
##                                -7.140e-01
##            reanalysis_dew_point_temp_k.l1
##                                 2.376e+00
## reanalysis_specific_humidity_g_per_kg.l1
##                                -1.523e+00
##                                     const
##                                -6.704e+02
##                                     trend
##                                 3.001e-04
```

Model Evaluation

Neither the `forecast::tsCV()` function nor the `greybox::ro()` function can be used to cross validate a multivariate time series model, so I wrote a for loop to evaluate this model using the forecast evaluation on a rolling origin method (500 origins). I forecast at three horizons: 1 week ahead, 6 weeks ahead, and 6 months ahead. With an MAE of 6.5 for a 1 week horizon, 14.5 for 6 weeks ahead, and 30.9 for 6 months ahead, this model is not as good as the Dynamic Regression, in spite of the fact that it includes more variables.

```r
# Divide time series with 500 observations in test set
train1 <- subset(ts.selected, end = 436) # subset of series ending at this point
test1 <- subset(ts.selected, start = 437) # subset of series beginning at next point

# Horizon = 1 week
h <- 1 # the horizon
n <- length(test1[,1]) - h + 1 # number of obs in test set (500) - horizon (1) + 1
fcmat <- matrix(0, nrow=n, ncol=h) # a matrix of 0s that is 500x1

for(i in 1:n)
{
  x <- subset(ts.selected, end = 436 + (i-1)) # the ts subset (for each iteration)
  refit <- VAR(x, p=1, type="both") # fit the ts subset
  fcmat[i,] <- forecast(refit, h=h)$forecast$total_cases[["mean"]] # forecast, extract the point foreca
}

# Calculate accuracy
print("1 week horizon accuracy")
```

```
## [1] "1 week horizon accuracy"
```

```r
accuracy(fcmat[,1], test1[,"total_cases"]) # compare forecasts to test set
```

```
##                    ME     RMSE      MAE  MPE MAPE       ACF1 Theil's U
## Test set -0.09551088 9.846422 6.484304 -Inf  Inf -0.1204734       NaN
```

```r
## Horizon = 6 weeks

h <- 6 # the horizon
n <- length(test1[,1]) - h + 1 # number of obs in test set (500) - horizon (6) + 1
fcmat <- matrix(0, nrow=n, ncol=h) # a matrix of 0s that is 495 x 6

for(i in 1:n)
{
  x <- subset(ts.selected, end = 436 + (i-1)) # the ts subset (for each iteration)
  refit <- VAR(x, p=1, type="both") # fit the ts subset
  fcmat[i,] <- forecast(refit, h=h)$forecast$total_cases[["mean"]] # forecast, extract the point foreca
}

# Calculate accuracy
print("6 week horizon accuracy")
```

```
## [1] "6 week horizon accuracy"
```

```r
accuracy(fcmat[,6], subset(ts.selected, start = 442)[,"total_cases"]) # compare forecasts to true value
```

```
##                 ME    RMSE     MAE MPE MAPE      ACF1 Theil's U
## Test set -1.036617 21.7324 14.5005 Inf  Inf 0.8037651       NaN
```

```r
## Horizon = 6 months

h <- 26 # the horizon
n <- length(test1[,1]) - h + 1 # number of obs in test set (500) - horizon (26) + 1
fcmat <- matrix(0, nrow=n, ncol=h) # a matrix of 0s that is 475 x 6
```

```r
for(i in 1:n)
{
  x <- subset(ts.selected, end = 436 + (i-1)) # the ts subset (for each iteration)
  refit <- VAR(x, p=1, type="both") # fit the ts subset
  fcmat[i,] <- forecast(refit, h=h)$forecast$total_cases[["mean"]] # forecast, extract the point foreca
}

# Calculate accuracy
print("6 month horizon accuracy")
```

```
## [1] "6 month horizon accuracy"
```

```r
accuracy(fcmat[,26], subset(ts.selected, start = 462)[,"total_cases"]) # compare forecasts to true valu
```

```
##                 ME     RMSE      MAE  MPE MAPE      ACF1 Theil's U
## Test set -3.193966 42.06621 30.93167 -Inf  Inf 0.9476811         0
```