Streamflow Record Extension Example

Dave Lorenz

February 27, 2015

This example demonstrates the move.2 and related functions in the smwrStats package. The example retrieves data from NWISweb using functions in the dataRetrieval package. The Data are from the North Fork Yellow Bank River near Odessa, Minn. (USGS station identifier 05292704) and the Yellow Bank River near Odessa, Minn. (USGS station identifier 05293000).

```
> # Load the smwrStats and dataRetrieval packages
> library(smwrStats)
> library(dataRetrieval)
> # Get the datasets and rename columns
> NFYB <- readNWISdv("05292704", parameterCd="00060", startDate="2000-10-01",
+   endDate="2001-09-30")
> NFYB <- renameNWISColumns(NFYB)
> YB <- readNWISdv("05293000", parameterCd="00060", startDate="2000-10-01",
+   endDate="2001-09-30")
> YB <- renameNWISColumns(YB)
```

1 The Lognormal Model

The objective of this example is to estimate the missing streamflow during water-year 2001 for the Yellow Bank River near Odessa, Minn. from the complete record at the upsteam site, North Fork Yellow Bank River near Odessa, Minn., based on only the water year 2001 data.

The first step is to merge the data by date, then build the MOVE.2 model. Because there is only a short distance between the gaging stations, a lag of 0 (the default) will be used in the model. The `merge` function is set up to create the two streamflow columns, Flow.NFYB and Flow.YB.

```
> # Merge the data
> YBM <- merge(NFYB, YB, by="Date", all=TRUE, suffixes=c(".NFYB", ".YB"))
> # Construct and print the model.
> YBM.m2ln <- move.2(Flow.YB ~ Flow.NFYB, data=YBM, distribution="lognormal")
> print(YBM.m2ln)
```

```
Call:
move.2(formula = Flow.YB ~ Flow.NFYB, data = YBM, distribution = "lognormal")

Coefficients:
   (Intercept) log(Flow.NFYB)
        0.8123         0.9070

Statistics of the variables:
Response (log(Flow.YB)):
mean.cor   sd.cor
   2.315    2.138
Predictor (log(Flow.NFYB)):
 mean    sd
1.656 2.358
Correlation coefficient: 0.993
                p-value: 0

Concurrent record length: 176
  Extended record length: 189
```

The statistics from the printed output indicate the that data are very highly correlated, 0.933 and the diagnostic plot, shown in figure 1, indicates a reasonably linear fit.

```
> # setSweave is a specialized function that sets up the graphics page for
> # Sweave scripts. For interactive use, it should be removed and the
> # default setting for set.up can be used.
> setSweave("graph01", 6, 6)
> plot(YBM.m2ln, which=2, set.up=FALSE)
> # Required call to close PDF output graphics
> graphics.off()
```

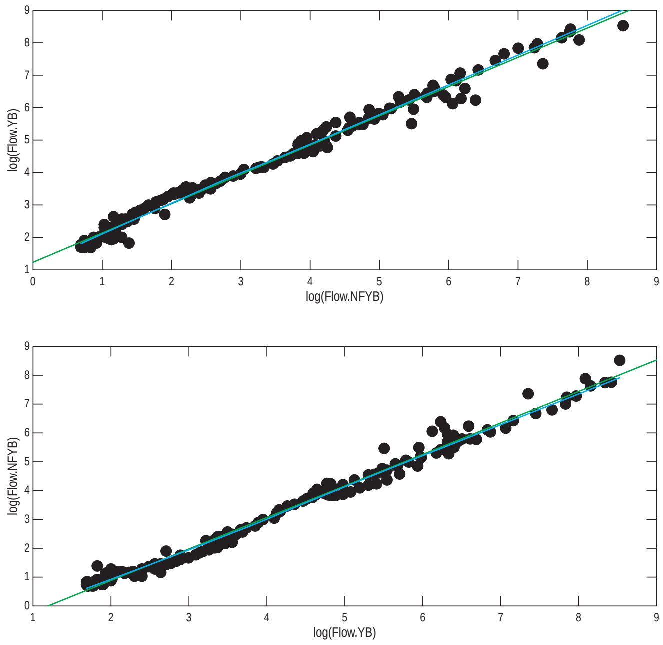

Figure 1. The diagnsotic plot showing the linearity of the fit.

The predicted values are computed using the `predict` function without specifying the `newdata` argument. The plot shows how well the concurrent data are predicted; it suggests that the concurrent values are underpredicted!

```
> # Predict all values
> YBM$Pred.ln <- predict(YBM.m2ln)
> setSweave("graph02", 6, 6)
> AA.pl <- with(YBM, timePlot(Date, Pred.ln, yaxis.log=TRUE))
> AA.pl <- with(YBM, addXY(Date, Flow.YB,
+   Plot=list(what="lines", color="green"), current=AA.pl))
> graphics.off()
```

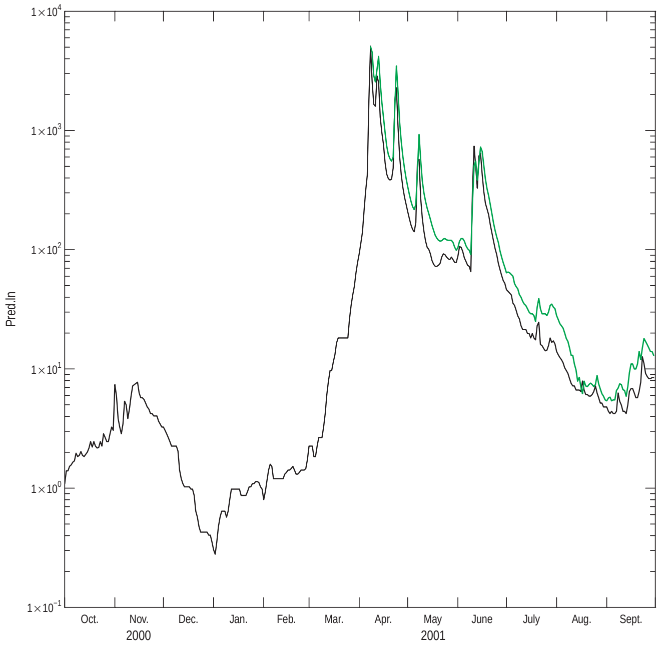

Figure 2. The predicted and concurrent values.

2 The Box-Cox Model

Moog and others (1999) suggest that power transforms other than log can be useful in streamflow record extension. The `optimBoxCox` function can be used to find good power transformations to make data more multivariate normal. The output from `optimBoxCox` can be used as the value for the `distribution` argument. The steps in building a Box-Cox transformation model are shown in the R code below.

The `move.2` function uses the rounded lambda values (Rnd. Lambda) rather than the optimized values. The graph indicates a much better fit to the concurrent values than the lognormal model.

```
> # Construct and print the power transforms for multivariate normality
> YBM.bc <- optimBoxCox(YBM[c("Flow.YB", "Flow.NFYB")])
> print(YBM.bc)

Optimized Box-Cox Transformations to Multinormality

          Est. Lambda Std.Err. Rnd. Lambda
Flow.YB       -0.0843   0.0240        -0.1
Flow.NFYB     -0.1118   0.0248        -0.1

> # Construct and print the model.
> YBM.m2bc <- move.2(Flow.YB ~ Flow.NFYB, data=YBM, distribution=YBM.bc)
> print(YBM.m2bc)

Call:
move.2(formula = Flow.YB ~ Flow.NFYB, data = YBM, distribution = YBM.bc)

Coefficients:
                 (Intercept) boxCox(Flow.NFYB,-0.1,26.46)
                     110.293                        2.281

Statistics of the variables:
Response (boxCox(Flow.YB,-0.1,66.03)):
mean.cor   sd.cor
   219.3    159.0
Predictor (boxCox(Flow.NFYB,-0.1,26.46)):
 mean    sd
47.78 69.69
Correlation coefficient: 0.993
                p-value: 0

Concurrent record length: 176
  Extended record length: 189
```

```
> # Predict all values
> YBM$Pred.bc <- predict(YBM.m2bc)
> setSweave("graph03", 6, 6)
> AA.pl <- with(YBM, timePlot(Date, Pred.bc, yaxis.log=TRUE))
> AA.pl <- with(YBM, addXY(Date, Flow.YB,
+   Plot=list(what="lines", color="green"), current=AA.pl))
> graphics.off()
```

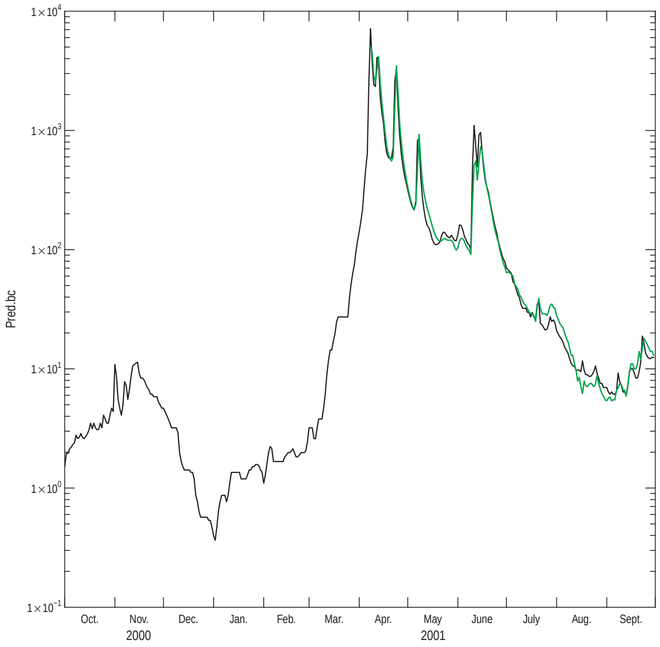

Figure 3. The predicted and concurrent values for the Box-Cox power transformation model.

References

[1] Moog, D.B., Whiting, P.J., and Thomas, R.B., 1999, Streamflow record extension using power transformations and applications to sediment transport: Water Resources Research, v. 35, p 243–254.