Using EGRET Data in a rloadest Model

Dave Lorenz

July 26, 2017

This example illustrates how to set up and use data retrieved and processed for an EGRET (Hirsch and De Cicco, 2015) in a rloadest load model. EGRET includes the statistical algorithm Weighted Regressions on Time, Discharge, and Season (WRTDS) that can compute loads and concentrations. WRTDS uses locally weighted regression on linear time, linear flow (discharge), and the first-order sine and cosine terms to model constituent concentrations and fluxes over time and through the range for flow.

This example uses the processed data supplied in the EGRET package, but any data retrieved and processed by the *readNWISDaily*, *readNWISSample*, *readNWISInfo* and *mergeReport* functions in EGRET can be used. The sullied data are nitrate plus nitrite data collected in the Choptank River near Greensboro, Maryland (USGS site identifier 01491000).

```
> # Load the necessary packages and the data
> library(survival) # required for Surv
> library(rloadest)
> library(EGRET)
> # Get the QW and daily flow data
> Chop.QW <- Choptank_eList$Sample
> Chop.Q <- Choptank_eList$Daily
```

1 Compute the Initial rloadest Model

The 7-parameter model (model number 9) is a typical model for relatively long-term records, longer than about 7 years and can be a good starting point for building a good model. The water-quality data in the Sample dataset for EGRET is stored in four columns—the minimum value, maximum value, an indicator of censoring, and the average value. That format can be converted to a valid response variable for *loadReg* using either *as.mcens* or *Surv*; *Surv* is preferred because if the data are uncensored or left-censored, then the "AMLE" method is used rather that the "MLE" method, which is always used with a response variable of class "mcens."

```
> # Compute the 7-parameter model.
> Chop.lr <- loadReg(Surv(ConcLow, ConcHigh, type="interval2") ~ model(9),
+   data=Chop.QW, flow="Q", dates="Date", conc.units="mg/L",
+   flow.units="cms", station="Choptank")
```

One of the first steps in assessing the fit is to look at the diagnostic plots for the linearity of the overall fit and each explanatory variable. The overall fit (figure 1) looks linear, but there are three low outliers and a tendency to larger scatter at larger predicted values

```
> # Plot the overall fit
> setSweave("graph01", 6, 6)
> plot(Chop.lr, which=1, set.up=FALSE)
> dev.off()

null device
          1
```

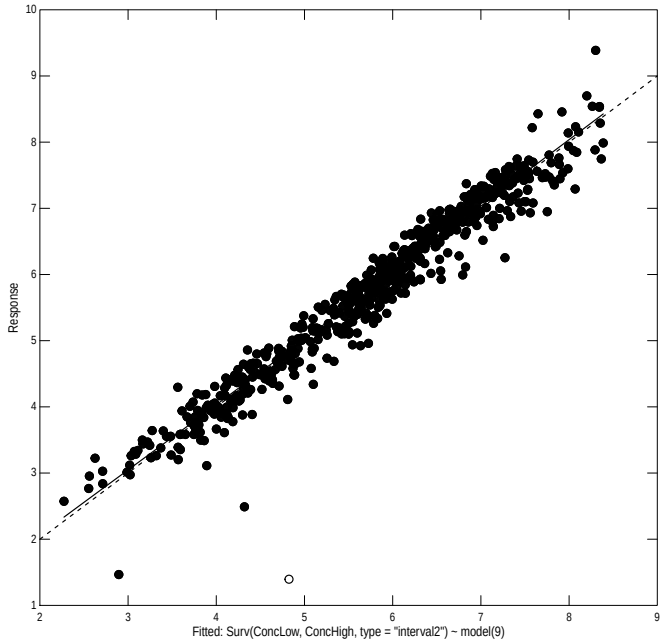

Figure 1. The overall fit.

The linearity of the explanatory variables is shown in figure 2. The partial residual plots for flow (lnQ and lnQ2) show nonlinearity in the second order (lnQ2). The partial residual plots for time (DECTIME and DECTIME2) show no nonlinearity, but the second-order term (DECTIME2) shows no trend and can therefore be removed from the model. The partial residual plots for seasonality (DECTIME and DECTIME2) show nonlinearity in both terms, suggesting the need for higher order seasonal terms.

```
> # Plot the explanatory variable fits
> setSweave("graph02", 6, 9)
> AA.lo <- setLayout(num.rows=3, num.cols=2)
> # Flow and flow squared
> setGraph(1, AA.lo)
> plot(Chop.lr, which="lnQ", set.up=FALSE)
> setGraph(2, AA.lo)
> plot(Chop.lr, which="lnQ2", set.up=FALSE)
> # Time and time squared
> setGraph(3, AA.lo)
> plot(Chop.lr, which="DECTIME", set.up=FALSE)
```

```
> setGraph(4, AA.lo)
> plot(Chop.lr, which="DECTIME2", set.up=FALSE)
> # Seasonality
> setGraph(5, AA.lo)
> plot(Chop.lr, which="sin.DECTIME", set.up=FALSE)
> setGraph(6, AA.lo)
> plot(Chop.lr, which="cos.DECTIME", set.up=FALSE)
> dev.off()

null device
          1
```

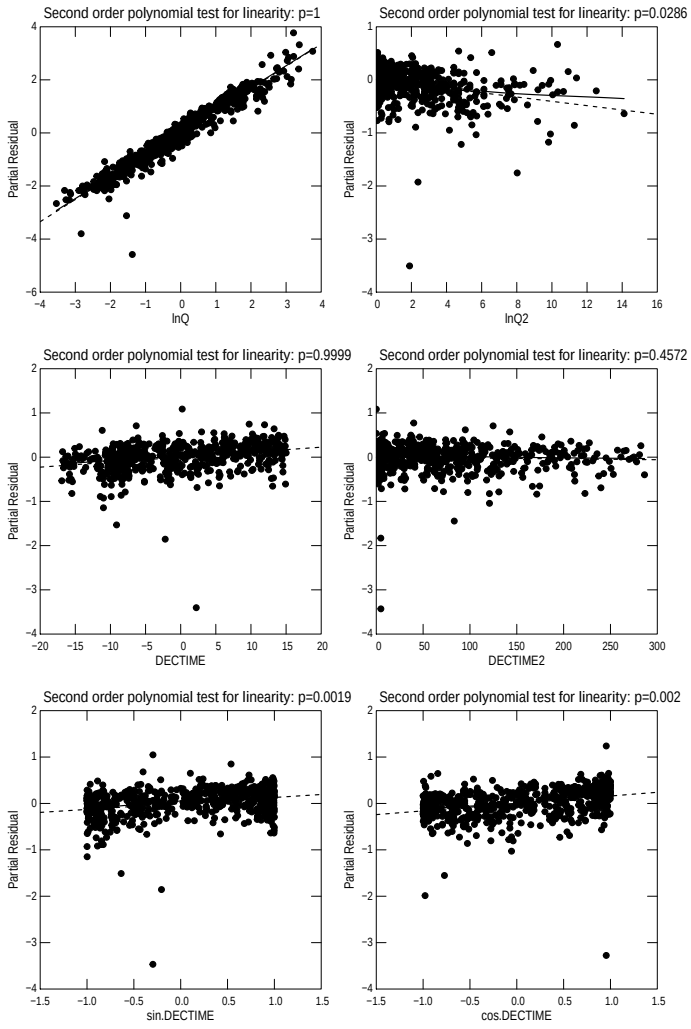

Figure 2. The linearity of the explanatory variables.

Figure 3 shows the relation between concentration and flow. The relation is not quadratic, but it appears that there is a distinct change at about 10 cubic meters per second. That relation can be modeled using piecewise linear, or segmented, terms.

```
> # Plot tconcentration and flow
> setSweave("graph03", 6, 6)
> # Use the average concentration (only one censored value)
```

```
> with(Chop.QW, xyPlot(Q, ConcAve, yaxis.log=TRUE, xaxis.log=TRUE))
> dev.off()

null device
          1
```

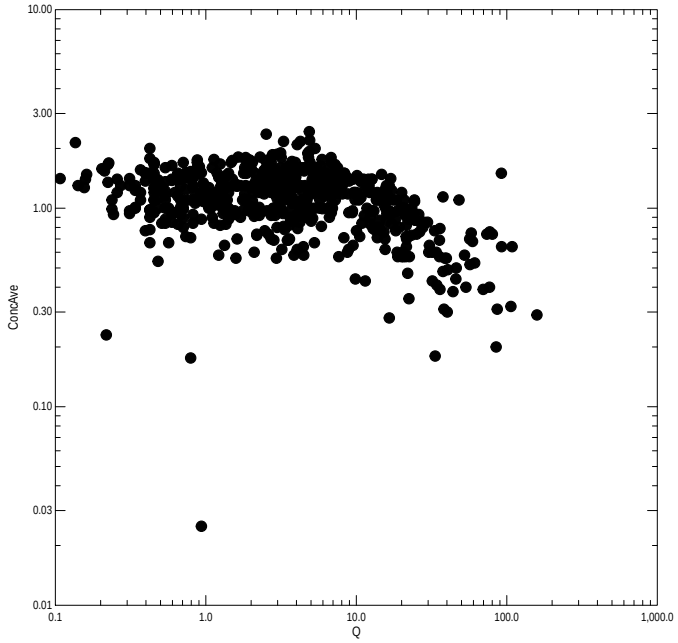

Figure 3. The relation between concentration and flow.

2 Construct the Modified rloadest Model

The *segLoadReg* can be used to build a piecewise linear model. It relies on the segmented package, which cannot model censored data to identify the breakpoints. For the first step censored values will be approximated by simple substitution; for the final model, the censored values are restored. One other quirk of *segLoadReg* is that the response term must be a variable, it cannot be constructed using *Surv* or any other function. Therefore, the breakpoint for this model will be identified using ConcAve, but the final model will be built using the censoring information.

```
> # Compute the breakpoint--the seg term must be the first term on
> # the right-hand side.
> Chop.lr <- segLoadReg(ConcAve ~ seg(LogQ, 1) + DecYear +
+     fourier(DecYear, 2),
+   data=Chop.QW, flow="Q", dates="Date", conc.units="mg/L",
+   flow.units="cms", station="Choptank")

Segmented regression results:
 AIC:
     lm      sm
405.344 334.227
Breakpoints (psi):
     Initial  Est. St.Err
psi1   1.211 1.994 0.1532

> # From the printed output, the breakpoint is 1.994 in natural log units,
> # about 7.4 cms
> # Compute and print the final model
> Chop.lr <- loadReg(Surv(ConcLow, ConcHigh, type="interval2") ~
+     segment(LogQ, 1.994) + DecYear + fourier(DecYear, 2),
+   data=Chop.QW, flow="Q", dates="Date", conc.units="mg/L",
+   flow.units="cms", station="Choptank")
> print(Chop.lr)

*** Load Estimation ***

Station: Choptank
Constituent: Surv(ConcLow, ConcHigh, type = "interval2")

           Number of Observations: 606
Number of Uncensored Observations: 605
           Center of Decimal Time: 1996.735
                  Center of ln(Q): 1.3098
```

```
                Period of record: 1979-10-24 to 2011-09-29

Selected Load Model:
--------------------

Surv(ConcLow, ConcHigh, type = "interval2") ~ segment(LogQ, 1.994) +
    DecYear + fourier(DecYear, 2)

Model coefficients:
                             Estimate Std. Error  z-score
(Intercept)                -17.886811   2.938636 -6.08677
segment(LogQ, 1.994)X        0.948716   0.018697 50.74174
segment(LogQ, 1.994)U1      -0.338732   0.040304 -8.40445
segment(LogQ, 1.994)P1       0.001323   0.050014  0.02646
DecYear                      0.011303   0.001473  7.67280
fourier(DecYear, 2)sin(k=1)  0.113551   0.021330  5.32351
fourier(DecYear, 2)cos(k=1)  0.143342   0.018775  7.63469
fourier(DecYear, 2)sin(k=2)  0.057079   0.017424  3.27584
fourier(DecYear, 2)cos(k=2)  0.060241   0.017818  3.38097
                            p-value
(Intercept)                   0e+00
segment(LogQ, 1.994)X         0e+00
segment(LogQ, 1.994)U1        0e+00
segment(LogQ, 1.994)P1        1e+00
DecYear                       0e+00
fourier(DecYear, 2)sin(k=1)   0e+00
fourier(DecYear, 2)cos(k=1)   0e+00
fourier(DecYear, 2)sin(k=2)   1e-03
fourier(DecYear, 2)cos(k=2)   7e-04

AMLE Regression Statistics
Residual variance: 0.09136
Generalized R-squared: 94.75 percent
G-squared: 1786 on 8 degrees of freedom
P-value: <0.0001
Prob. Plot Corr. Coeff. (PPCC):
  r = 0.9608
  p-value = 0
Serial Correlation of Residuals: 0.2493

Variance Inflation Factors:
                              VIF
segment(LogQ, 1.994)X       4.647
segment(LogQ, 1.994)U1      3.490
segment(LogQ, 1.994)P1      3.373
DecYear                     1.032
```

```
fourier(DecYear, 2)sin(k=1) 1.499
fourier(DecYear, 2)cos(k=1) 1.165
fourier(DecYear, 2)sin(k=2) 1.048
fourier(DecYear, 2)cos(k=2) 1.010

Comparison of Observed and Estimated Loads
------------------------------------------
      Summary Stats: Loads in kg/d
------------------------------------------
      Min 25% 50% 75%  90%  95%   Max
Est 12.70 122 364 920 1660 2090  4350
Obs  2.02 116 352 911 1670 1990 11900

Bias Diagnostics
----------------
 Bp: -0.4405 percent
PLR: 0.9956
  E: 0.7596
```

This segmented model has three variables— with names ending in X, U1, and P1. The coefficient for the variable ending in X is the slope for the variable less that the breakpoint, the coefficient for the variable ending in U1 is the change in slope above the breakpoint, and the coefficient for the variable ending in P1 should always be close to 0.

No partial residual plot indicates any nonlinearity. Figure 4 shows 5 selected partial residual plots.

```
> # Plot the explanatory variable fits
> setSweave("graph04", 6, 9)
> AA.lo <- setLayout(num.rows=3, num.cols=2)
> # Segmented flow
> setGraph(1, AA.lo)
> plot(Chop.lr, which="segment(LogQ, 1.994)X", set.up=FALSE)
> setGraph(2, AA.lo)
> plot(Chop.lr, which="segment(LogQ, 1.994)U1", set.up=FALSE)
> # Time
> setGraph(3, AA.lo)
> plot(Chop.lr, which="DecYear", set.up=FALSE)
> # Seasonality
> setGraph(5, AA.lo)
> plot(Chop.lr, which="fourier(DecYear, 2)sin(k=2)", set.up=FALSE)
> setGraph(6, AA.lo)
> plot(Chop.lr, which="fourier(DecYear, 2)cos(k=2)", set.up=FALSE)
> dev.off()
```

```
null device
          1
```

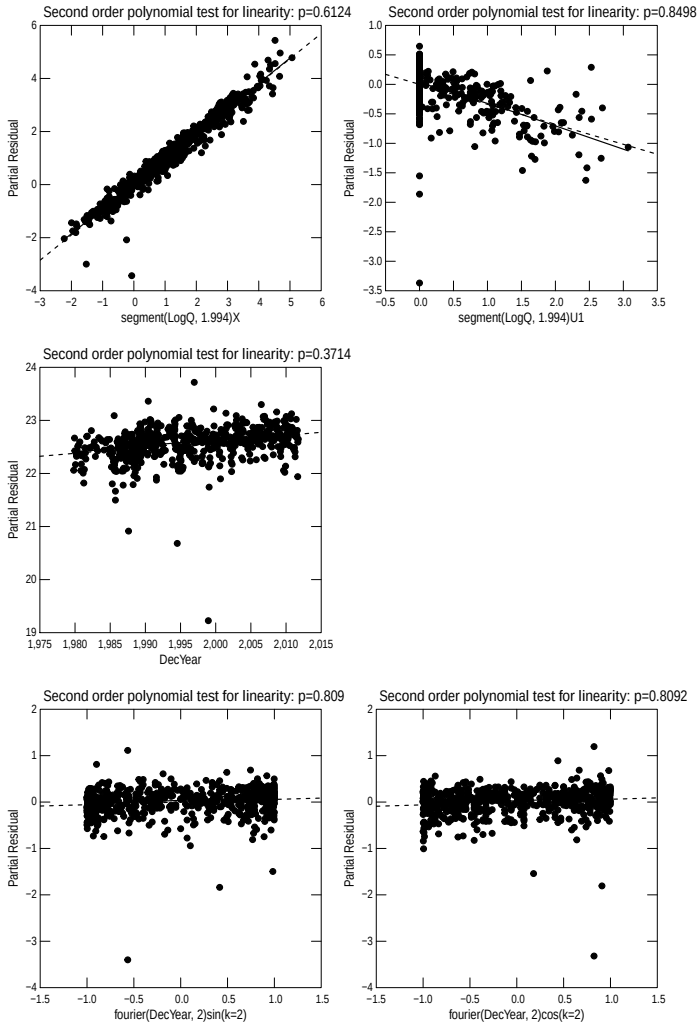

Figure 4. Partial residual plots.

3 Further Considerations

Although not specifically stated by Runkel and others (2004), rloadest is designed to model constituent loads over time periods from about 2 to 10 years. The 2-year minimum is a practical limit based on a minimum number of samples required to build a good model. The 10-year period is based on the application by Cohn (personal communication, February 2006) for Chesapeake Bay load estimates. For time periods longer than about 10 years, both trends and the flow-concentration relation can require additional modeling tools (Vecchia, 2000). WRTDS accounts for those long-term changes by using locally weighted regression.

To determine if there are unmodeled flow dependencies in the load model, the mean residual by water year can be compared to the mean flow by water year. The R code immediately following this paragraph computes those means and graphs the relation over time. There appears to be a strong correlation between the mean residual, in black, and the mean annual flow, in red. This relation is different from the relation between the residual and observed flow, which is correctly modeled as indicated by the partial residual plots.

```
> # Compute the mean residual and flow by water year
> MeanRes <- tapply(residuals(Chop.lr), waterYear(Chop.QW$Date), mean)
> MeanQ <- with(Chop.Q, tapply(LogQ, waterYear(Date), mean))
> # Get the years and convert the means to scaled vectors (for plotting)
> MeanWY <- as.integer(names(MeanQ))
> MeanRes <- as.vector(scale(MeanRes))
> MeanQ <- as.vector(scale(MeanQ))
> # Plot them
> setSweave("graph05", 6, 6)
> AA.pl <- timePlot(MeanWY, MeanRes, Plot=list(what="overlaid"),
+   yaxis.range=c(-2.5, 2.5))
> addXY(MeanWY, MeanQ, Plot=list(what="overlaid", color="red"),
+   current=AA.pl)
> dev.off()

null device
          1
```

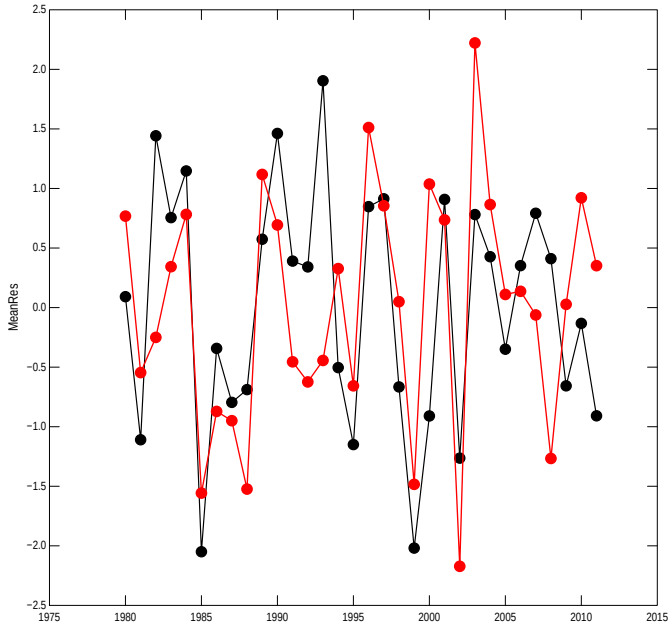

Figure 5. Mean residuals and mean flow.

Streamflow anomalies, described by Vecchia (2000), cannot be used directly in this model because it is a segmented model. For an example where flow anomalies can be used, see the `Application 3, Analysis of an Censored Constituent using a Seasonal Model`. For these data, a flow dependence can be computed that incorporates the average log flow for a previous period of time. The R code immediately following this paragraph, retrieves the flow data for 1 year prior to the beginning of the analysis period and computes the mean log flow for 3-, 6-, 9-, and 12-months. The `movingAve` function in `smwrBase` is used to compute those flow dependencies. The `mergeQ` function in `smwrBase` is used to extract the computed columns into the calibration dataset. The largest correlation between the residuals and the flow dependencies is the 3-month period, but the 12-month is nearly as large.

```
> # Retrieve the flow data , beginning 1978-10-01, and compute log flowe in cms
> Chop.ExQ <- renameNWISColumns(readNWISdv(
+   "01491000", "00060", "1978-10-01", "2011-09-30"))
> Chop.ExQ$LogQ <- log(Chop.ExQ$Flow/35.31467)
> # Compute the Dependencies
> Chop.ExQ <- transform(Chop.ExQ,
```

```
+   Dep3mo=movingAve(LogQ, 91, pos="trailing"),
+   Dep6mo=movingAve(LogQ, 182, pos="trailing"),
+   Dep9mo=movingAve(LogQ, 273, pos="trailing"),
+   Dep12mo=movingAve(LogQ, 365, pos="trailing"))
> # Merge the dependencies into the calibration dataset
> Chop.ExQW <- mergeQ(Chop.QW, FLOW=c("Dep3mo", "Dep6mo", "Dep9mo", "Dep12mo"),
+   DATES="Date", Qdata=Chop.ExQ, Plot=FALSE)
> # Which has the largest correlation?
> cor(residuals(Chop.lr), Chop.ExQW[c("Dep3mo", "Dep6mo", "Dep9mo", "Dep12mo")])
```

```
        Dep3mo    Dep6mo    Dep9mo   Dep12mo
[1,] 0.1285903 0.1155899 0.1189847 0.1267015
```

One more consideration for factors that affect flow is the hysteresis in concentration between rising and falling flows. The `hysteresis` function in `smwrBase` can be used to construct a variable that can model differences in concentration between rising and falling limbs of the hydrograph. The R code immediately following this paragraph computes that hysteresis based on a 1 day time step, which would be appropriate for this small basin. The correlation is quite large, so it should be included in the model.

```
> # Compute the Hysterisis and merge into the new calibration data set
> Chop.ExQ <- transform(Chop.ExQ, Hy1=hysteresis(LogQ, 1))
> # Merge into the calibration dataset
> Chop.ExQW <- mergeQ(Chop.ExQW, FLOW="Hy1",
+   DATES="Date", Qdata=Chop.ExQ, Plot=FALSE)
> # Is it correlated with the residuals?
> cor(residuals(Chop.lr), Chop.ExQW$Hy1)
```

```
[1] 0.1242115
```

One final possibility, nonlinearities in the trend, must be considered. For some data the nonlinearities are very apparent, but they are more subtle for these data. There can be many approaches to describing the nonlinearities in trend: a second-order polynomial, using the `quadratic` function in `smwrBase`; higher-order polynomials using `poly` in `base`; piecewise linear terms using `trends` in `smwrStats`; and piecewise curvilinear terms using `curvi` in `smwrStats`. For most data, the partial residual plot over time would suggest any nonlinearity. Figure 6 shows that these data are very noisy, indicate very little deviation from a linear trend, but show the possibility of some leveling off in trend for a short period around 1995. Possible approaches for modeling the trend include piecewise linear with breaks around 1991 and 1998, curvilinear trends with midpoints at 1980 and 2010 with half-widths of 15 years (meeting in 1995), and a simple linear trend. For this example, the

simple linear trend was selected because it was the simplest model, the absolute values of the coefficient of variation of the jackknife bias were all less than 0.25, and had good performance on the bias statistics. The code for the jackknife estimates is shown in the box below, but not executed.

```
# The simple linear trend
jackStats(loadReg(Surv(ConcLow, ConcHigh, type="interval2") ~
segment(LogQ, 1.994) + DecYear +
fourier(DecYear, 2) + Hy1 + Dep3mo,
data=Chop.ExQW, flow="Q", dates="Date", conc.units="mg/L",
flow.units="cms", station="Choptank"))
# The piecewise linear trend
jackStats(loadReg(Surv(ConcLow, ConcHigh, type="interval2") ~
segment(LogQ, 1.994) + trends(DecYear, c(1991, 1998)) +
fourier(DecYear, 2) + Hy1 + Dep3mo,
data=Chop.ExQW, flow="Q", dates="Date", conc.units="mg/L",
flow.units="cms", station="Choptank"))
# And the curvilinear trend
jackStats(loadReg(Surv(ConcLow, ConcHigh, type="interval2") ~
segment(LogQ, 1.994) + curvi(DecYear, c(1980, 15, 2010, 15)) +
fourier(DecYear, 2) + Hy1 + Dep3mo,
data=Chop.ExQW, flow="Q", dates="Date", conc.units="mg/L",
flow.units="cms", station="Choptank"))
```

```
> # Compute the extended load regression exluding time
> Chop.lrEx <- loadReg(Surv(ConcLow, ConcHigh, type="interval2") ~
+     segment(LogQ, 1.994) +
+     fourier(DecYear, 2) + Hy1 + Dep3mo,
+   data=Chop.ExQW, flow="Q", dates="Date", conc.units="mg/L",
+   flow.units="cms", station="Choptank")
> # Use the functions in smwrGraphs to easily add the smooth line
> # Plot the residuals over time (decimal year)
> # zooming in to the bulk of the residuals
> setSweave("graph06", 6, 6)
> AA.pl <- xyPlot(Chop.QW$DecYear, residuals(Chop.lrEx),
+   xtitle="Decimal Time", ytitle="Partial Residual",
+   yaxis.range=c(-1,1))
> # Add a smmothed line, setting family to "gaussian"--better for regression
> addSmooth(AA.pl, family="gaussian")
> dev.off()

null device
          1
```

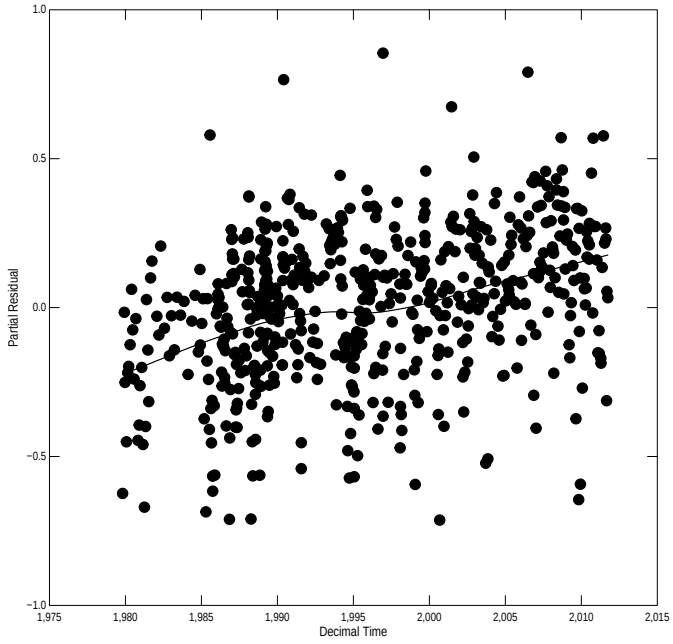

Figure 6. The temporal residual trend with smoothed line.

The extended model can now be built. The R code following this paragraph extends the model created in the previous section by adding hysteresis, the flow dependency terms, and the linear trend term. The printed result indicates that all terms, are significant at the 0.05 alpha level and the bias diagnostics indicate very good agreement between the observed and estimated loads. The diagnostic plots are not shown; the q-normal residual plot shows a lack of agreement with the normal distribution as indicated by the PPCC section of the report; the correlogram indicates serial correlation, which agrees with the printed report; and the partial residual plots show no issues, with the possible exception of some nonlinearity with the hysteresis term(Hy1).

```
> # Compute and print the Extended model
> Chop.lrEx <- loadReg(Surv(ConcLow, ConcHigh, type="interval2") ~
+     segment(LogQ, 1.994) + trends(DecYear, c(1991, 1998)) +
+     fourier(DecYear, 2) + Hy1 + Dep3mo,
+   data=Chop.ExQW, flow="Q", dates="Date", conc.units="mg/L",
+   flow.units="cms", station="Choptank")
> print(Chop.lrEx)
```

```
*** Load Estimation ***

Station: Choptank
Constituent: Surv(ConcLow, ConcHigh, type = "interval2")

           Number of Observations: 606
Number of Uncensored Observations: 605
           Center of Decimal Time: 1996.735
                  Center of ln(Q): 1.3098
                Period of record: 1979-10-24 to 2011-09-29

Selected Load Model:
--------------------

Surv(ConcLow, ConcHigh, type = "interval2") ~ segment(LogQ, 1.994) +
    trends(DecYear, c(1991, 1998)) + fourier(DecYear, 2) + Hy1 +
    Dep3mo

Model coefficients:
                                      Estimate
(Intercept)                          4.3212153
segment(LogQ, 1.994)X                0.8793154
segment(LogQ, 1.994)U1              -0.2999984
segment(LogQ, 1.994)P1              -0.0009728
trends(DecYear, c(1991, 1998))1979-1991  0.0273868
trends(DecYear, c(1991, 1998))1991-1998 -0.0092955
trends(DecYear, c(1991, 1998))1998-2012  0.0172429
fourier(DecYear, 2)sin(k=1)          0.0472512
fourier(DecYear, 2)cos(k=1)          0.2107569
fourier(DecYear, 2)sin(k=2)          0.0708631
fourier(DecYear, 2)cos(k=2)          0.0438938
Hy1                                  0.1233410
Dep3mo                               0.1405516
                                     Std. Error  z-score
(Intercept)                            0.055617 77.69572
segment(LogQ, 1.994)X                  0.020170 43.59598
segment(LogQ, 1.994)U1                 0.039268 -7.63978
segment(LogQ, 1.994)P1                 0.048141 -0.02021
trends(DecYear, c(1991, 1998))1979-1991   0.006009  4.55748
trends(DecYear, c(1991, 1998))1991-1998   0.006213 -1.49616
trends(DecYear, c(1991, 1998))1998-2012   0.003891  4.43150
fourier(DecYear, 2)sin(k=1)            0.023589  2.00313
fourier(DecYear, 2)cos(k=1)            0.020295 10.38442
fourier(DecYear, 2)sin(k=2)            0.016731  4.23552
fourier(DecYear, 2)cos(k=2)            0.017155  2.55868
Hy1                                    0.024861  4.96114
```

```
Dep3mo                                    0.021300  6.59866
                                         p-value
(Intercept)                               0.0000
segment(LogQ, 1.994)X                     0.0000
segment(LogQ, 1.994)U1                    0.0000
segment(LogQ, 1.994)P1                    0.9825
trends(DecYear, c(1991, 1998))1979-1991   0.0000
trends(DecYear, c(1991, 1998))1991-1998   0.1296
trends(DecYear, c(1991, 1998))1998-2012   0.0000
fourier(DecYear, 2)sin(k=1)               0.0438
fourier(DecYear, 2)cos(k=1)               0.0000
fourier(DecYear, 2)sin(k=2)               0.0000
fourier(DecYear, 2)cos(k=2)               0.0101
Hy1                                       0.0000
Dep3mo                                    0.0000

AMLE Regression Statistics
Residual variance: 0.08297
Generalized R-squared: 95.26 percent
G-squared: 1848 on 12 degrees of freedom
P-value: <0.0001
Prob. Plot Corr. Coeff. (PPCC):
  r = 0.9561
  p-value = 0
Serial Correlation of Residuals: 0.2382

Variance Inflation Factors:
                                           VIF
segment(LogQ, 1.994)X                   5.955
segment(LogQ, 1.994)U1                  3.648
segment(LogQ, 1.994)P1                  3.441
trends(DecYear, c(1991, 1998))1979-1991 1.727
trends(DecYear, c(1991, 1998))1991-1998 2.909
trends(DecYear, c(1991, 1998))1998-2012 1.971
fourier(DecYear, 2)sin(k=1)             2.018
fourier(DecYear, 2)cos(k=1)             1.499
fourier(DecYear, 2)sin(k=2)             1.063
fourier(DecYear, 2)cos(k=2)             1.031
Hy1                                     1.296
Dep3mo                                  2.790

Comparison of Observed and Estimated Loads
------------------------------------------
      Summary Stats: Loads in kg/d
------------------------------------------
      Min 25% 50% 75%  90%  95%   Max
```

```
Est 12.10 124 376 928 1610 2140  4970
Obs  2.02 116 352 911 1670 1990 11900

Bias Diagnostics
----------------
 Bp: -0.1761 percent
PLR: 0.9982
  E: 0.7852
```

4 WRTDS and rloadest Model Comparisons

A thorough comparison of the WRTDS and rloadest models is beyond the scope of this vignette, but it is possible to find some commonalities and differences between the models. The first step is top compute the residuals for the WRTDS model. For simplicity, the average concentration is used, which has very little effect for the one censored value. The R code following this paragraph computes the residuals and plots the water-year series of box plots for the WRTDS, modified, and extended load models.

```
> # Compute the WRTDS residuals and the water year
> Chop.QW <- transform(Chop.QW, Res=log(ConcAve) - yHat,
+                      WY=waterYear(Date, numeric=FALSE))
> # Graph the residuals
> setSweave("graph07", 6, 9)
> AA.lo <- setLayout(num.rows=3, shared.x=1)
> # The WRTDS residuals over time
> AA.mr <- setGraph(1, AA.lo)
> with(Chop.QW, boxPlot(Res, group=WY, Box=list(show.counts=FALSE),
+   yaxis.range=c(-1,1), xlabels.rotate=TRUE, margin=AA.mr))
> refLine(horizontal=0)
> addTitle(Main="WRTDS", Position="inside", Bold=FALSE)
> # Modified residuals over time
> AA.mr <- setGraph(2, AA.lo)
> with(Chop.QW, boxPlot(residuals(Chop.lr), group=WY,
+   Box=list(show.counts=FALSE),
+   yaxis.range=c(-1,1), xlabels.rotate=TRUE, margin=AA.mr))
> refLine(horizontal=0)
> addTitle(Main="Modified", Position="inside", Bold=FALSE)
> # Extended residuals over time
> AA.mr <- setGraph(3, AA.lo)
> with(Chop.QW, boxPlot(residuals(Chop.lrEx), group=WY,
+   Box=list(show.counts=FALSE),
+   yaxis.range=c(-1,1), xlabels.rotate=TRUE, margin=AA.mr))
> refLine(horizontal=0)
> addTitle(Main="Extended", Position="inside", Bold=FALSE)
> dev.off()

null device
          1
```

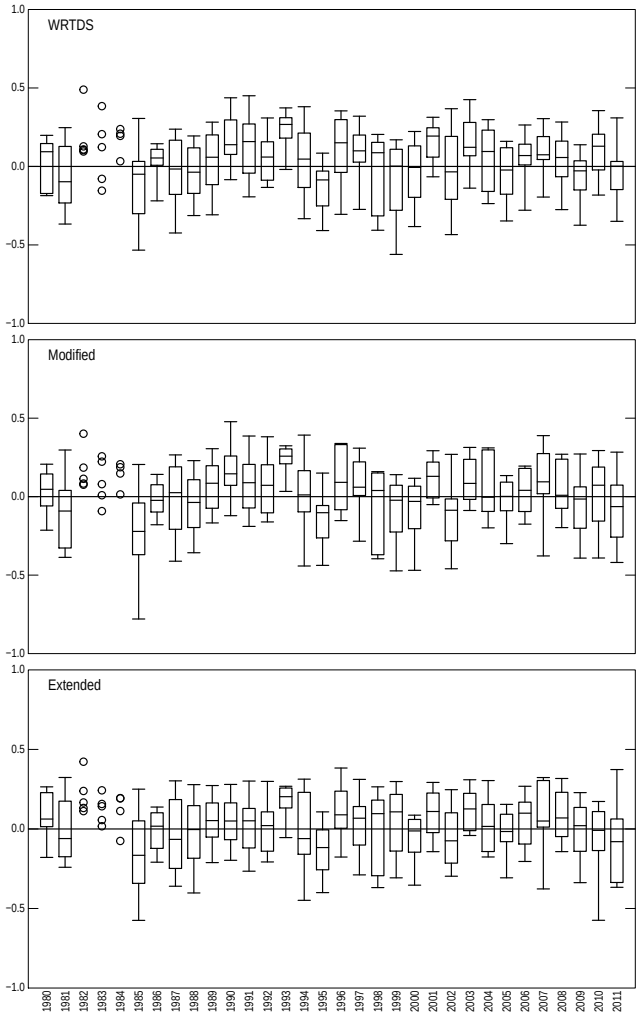

Figure 7. Residual box plots by water year for each of the models.

The pattern of the boxes, or individual values when there are five or fewer values in a water year, are very similar—the medians and ranges of the boxes are generally in the same direction from the 0 line among the models. This suggests that the WRTDS and the load models are very similar in their description of the concentrations and loads. But there is one major difference–the residuals of the extended model are much closer to the 0 line than the other two. The number of cases where the box lies outside of the 0 line or all individual values lie on one side is 11 for the WRTDS model 1982,

1984, 1986, 1990, 1993, 1995, 1997, 2001, 2003, 2006, and 2007; 8 for the modified model 1982, 1984, 1985, 1990, 1993, 1995, 1997, and 2007; but only 2 for the extended model 1982 and 1993. That suggests that the addition of flow dependency and hysteresis reduced annual bias in the model. However, there remain persistent patterns of bias—for example the median residual is greater than 0 for all models from 1989 through 1993, suggesting that concentration or load estimates for that time could be biased and that the model could be improved.

The extended load model uses 11 parameters: the intercept, three terms for flow, decimal time, four parameters for seasonality and the hysteresis and flow dependency terms. Cleveland and others (1996) describe a method to compute the equivalent number of parameters for local regression (the foundation for WRTDS). Applying their method to the WRTDS Choptank Nitrate plus Nitrite model results in the equivalent of 32.4 parameters, far more than were used in the extended load model.

Concentration and load modeling is both art and science. The knowledge of water chemistry and an understanding of the dynamics of the hydrologic system are required to paint a picture of the concentration or load. WRTDS can be expected to produce a good and useful model, but with extra effort, the modeling capabilities in the `rloadest` package can yield a better understanding of the dynamics. The practitioner must balance needs and effort.

References

[1] Cleveland, W.S., Grosse, E., and Shyu, M-J., 1996, A package of C and Fortran routines for fitting local regression models: Loess user's manual:Bell Labs, Technical Report. Available as PostScript report at http://www.stat.purdue.edu/ wsc/papers.html.

[2] Hirsch, R.M. and De Cicco, L.A., 2015, User guide to Exploration and Graphics for RivEr Trends (EGRET) and dataRetrieval R for hydrologic data (version 2.0, February 2015): U.S. Geological Survey Techniques and Methods book 4, chap A10, 93 p. Available at http://dx.doi.org/10.3133/tm4A10.

[3] Runkel, R.L., Crawford, C.G., and Cohn, T.A., 2004, Load estimator (LOADEST): a FORTRAN program for estimating constituent loads in streams and rivers: U.S. Geological Survey Techniques and Methods book 4, chap. A5, 69 p.

[4] Vecchia, A.V., 2000, Water-quality trend analysis and sampling design for the Souris River, Saskatchewan, North Dakota, and Manitoba: U.S. Geological Survey Water-Resources Investigations Report 00-4019, 77 p.