Regression

Dave Lorenz

December 30, 2015

Abstract

These examples demonstrate some of the functions and statistical methods for computing the regressions for censored response variables that are available in the `smwrQW` package.

Contents

1 Introduction

The examples in this vignette use the TCEReg dataset from the NADA package. The examples in this vignette use the function as.lcens to convert those data to a form used by the functions demonstrated; the class ”lcens” is most appropriate for these data as they are only left-censored and have only the value and an indicator of censoring. The functions demonstrated in these examples will also accept data of class ”qw.” The R code following this paragraph gets the data and creates a column named ”TCE” of class ”lcens.” Only logistic and maximum likelihood estimation methods are described in this vignette because those methods support multiple explanatory variables.

```
> # Load the smwrQW package
> library(smwrQW)
> # And the data
> data(TCEReg, package="NADA")
> # Convert the data to column TCE
> # For these data, force the reporting limit to be 1 unless the censoring
> # limit is specified.
> TCEReg <- transform(TCEReg, TCE=as.lcens(TCEConc, 1, censor.codes=TCECen))
```

2 Logistic regression

Logistic regression models a binary response variable as the probability of observing the larger value. For 0/1 coded data, that is the probability of observing 1, which represents exceeding a specified threshold value of the water-quality data. Helsel (2012) provides a brief introduction to logistic regression, other good references for logistic regression include McCullagh and Nedler (1999) and Harrel (2001).

The example below illustrates the recoding of values to 0/1, building the regression model and the detailed printed output from the `binaryReg` function that is in the `smwrStats` package (Lorenz, 2015). The output from the `binaryReg` function includes the summary information from the regression, two goodness-of-fit tests, four measure of predictive power, and influence diagnostics. Details of the output are presented in the ”Logistic” vignette in `smwrStats`.

```
> # Append a column of 0/1 values to the data
> # The maximum censored values is 5, which is the default criterion
> TCEReg <- cbind(TCEReg, with(TCEReg, code01(TCE01=TCE)))
> # Build the regression model
> TCE.lr <- glm(TCE01 ~ Depth + PopDensity, data=TCEReg,
+   family=binomial)
> # Detailed output from the logistic regression
> binaryReg(TCE.lr)


Call:
glm(formula = TCE01 ~ Depth + PopDensity, family = binomial,
    data = TCEReg)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.1805  -0.5010  -0.3943  -0.3485   2.5844

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -2.819673   0.535304  -5.267 1.38e-07 ***
Depth       -0.001319   0.001704  -0.774  0.43918
PopDensity   0.156350   0.051458   3.038  0.00238 **
---
Signif. codes:  0 Ś***Š 0.001 Ś**Š 0.01 Ś*Š 0.05 Ś.Š 0.1 Ś Š 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 182.69  on 246  degrees of freedom
Residual deviance: 169.92  on 244  degrees of freedom
AIC: 175.92

Number of Fisher Scoring iterations: 5


Likelihood ratio test: 12.7677 on 2 degrees of freedom, p-value is 0.0017

Response profile:
  TCE01 Response Counts
1     0        0    217
2     1        1     30

 Goodness of fit tests

        le Cessie-van Houwelingen GOF test

data:  TCE01 ~ Depth + PopDensity
Chisq = 24.023, df = 7.8078, p-value = 0.001997
alternative hypothesis: Some lack of fit
null hypothesis: No lack of fit
sample estimates:
       Q     E[Q]    se[Q]
41.08632 13.35357  6.75847

Distance between observations:
  maximum bandwidth
 4.301572  1.193963


        Hosmer-Lemeshow goodness of fit test

data:  TCE01 ~ Depth + PopDensity
Chi-square = 12, Number of groups = 10, p-value = 0.1
alternative hypothesis: Some lack of fit
null hypothesis: No lack of fit
sample estimates:
   Size Expected Counts
1    26    1.249      1
2    24    1.443      0
3    24    1.546      1
4    25    1.804      1
5    25    2.076      2
6    24    2.437      3
7    25    2.900      4
8    24    3.845      7
9    24    5.025      8
10   26    7.674      3


Predictive power estimates:
McFadden R-squared: 0.0699
adjusted R-squared: 0.048


Classification table.
Percent correct: (1 is sensitivity, 0 is specificity)
   1    0
 0.0 99.5
```

```
Concordance Index, based on 6510 pairs
Discordant       Tied Concordant
  27.51152    0.04608   72.44240

Area under the ROC curve: 0.725


Influence diagnostic test criteria:
leverage   cooksD    dfits
 0.04858  0.87272  0.49423
        Observations exceeding at least one test criterion
    TCE01    yhat   resids deviance.res pearson.res leverage   cooksD    dfits
56      0 0.44106 -0.44106      -1.0786     -0.8883 0.06367* 0.04160  -0.3539
67      0 0.43716 -0.43716      -1.0722     -0.8813 0.06398* 0.03075  -0.3040
70      0 0.50181 -0.50181      -1.1805     -1.0036 0.09537* 1.33263* -2.1712*
149     0 0.08627 -0.08627      -0.4248     -0.3073 0.05563* 0.03117  -0.3062
162     0 0.09052 -0.09052      -0.4356     -0.3155 0.04933* 0.03107  -0.3058
173     0 0.03008 -0.03008      -0.2471     -0.1761 0.02940  0.08329   0.5075*
175     0 0.07699 -0.07699      -0.4003     -0.2888 0.07054* 0.02587  -0.2786
243     0 0.08690 -0.08690      -0.4264     -0.3085 0.05468* 0.03125  -0.3066
```

The printed output indicates that the significance level of Depth as an explantory variable is much larger then 0.05 and the p-value from the le Cessie-van Houwelingen GOF test is much smaller than 0.05, suggesting a lack of fit. The code following this paragraph redoes the logistic regression, dropping Depth as an explanatory variable. In this case, the p-value from the le Cessie-van Houwelingen GOF test is larger than 0.05, suggesting no lack of fit. The performance of the model is similar to the previous mode and there are fewer observations that exceed at least one of the test criteria. Observation number 70 is printed; the concentration is censored at 1 and the population density (PopDensity) is 19, which is the largest value in the dataset. A graph of the observed data and the fitted line is created in the last part of the code.

```
> # Update the regression model
> TCE.lr <- update(TCE.lr, ~ . - Depth)
> # Detailed output from the logistic regression
> binaryReg(TCE.lr)


Call:
glm(formula = TCE01 ~ PopDensity, family = binomial, data = TCEReg)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.2342  -0.4852  -0.3803  -0.3502   2.3757

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -3.10111    0.41988  -7.386 1.52e-13 ***
PopDensity   0.17020    0.04933   3.450  0.00056 ***
---
Signif. codes:  0 Ś***Š 0.001 Ś**Š 0.01 Ś*Š 0.05 Ś.Š 0.1 Ś Š 1
```

```
(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 182.69  on 246  degrees of freedom
Residual deviance: 170.59  on 245  degrees of freedom
AIC: 174.59

Number of Fisher Scoring iterations: 5


Likelihood ratio test: 12.0987 on 1 degrees of freedom, p-value is 5e-04

Response profile:
  TCE01 Response Counts
1     0        0    217
2     1        1     30

 Goodness of fit tests

        le Cessie-van Houwelingen GOF test

data:  TCE01 ~ PopDensity
Chisq = 7.9802, df = 4.4442, p-value = 0.1189
alternative hypothesis: Some lack of fit
null hypothesis: No lack of fit
sample estimates:
        Q      E[Q]     se[Q]
26.696925 14.867468  9.973724

Distance between observations:
  maximum bandwidth
3.3896756 0.7799919

Too few unique predicted values for Hosmer-Lemeshow Test

Predictive power estimates:
McFadden R-squared: 0.0662
adjusted R-squared: 0.0553


Classification table.
Percent correct: (1 is sensitivity, 0 is specificity)
   1    0
 0.0 99.5

Concordance Index, based on 6510 pairs
Discordant       Tied Concordant
    22.611      9.539     67.849

Area under the ROC curve: 0.726
```

```
Influence diagnostic test criteria:
leverage   cooksD    dfits
 0.03644  0.84151  0.42801
        Observations exceeding at least one test criterion
   TCE01   yhat  resids deviance.res pearson.res leverage  cooksD    dfits
56     0 0.4482 -0.4482       -1.091     -0.9013 0.06452* 0.0385  -0.2776
67     0 0.4482 -0.4482       -1.091     -0.9013 0.06452* 0.0385  -0.2776
70     0 0.5331 -0.5331       -1.234     -1.0686 0.09042* 3.1459* -2.9067*
```

```
> # Print the farthest outlier
> TCEReg[70, ]
```

```
   TCECen TCEConc LandUse PopDensity PctIndLU Depth PopAbv1 TCE TCE01
70   TRUE       1       9         19        4   109       1  <1     0
```

```
> # Plot the data and fit
> setSweave("graph01", 6 ,6)
> # Create the graph,
> # first create a jittered column to see more individual points
> TCEReg <- transform(TCEReg,
+   TCEjit=TCE01 + runif(nrow(TCEReg), -.1, .1))
> with(TCEReg, xyPlot(PopDensity, TCEjit,
+   ytitle="Probability of exceeding criterion"))
> # Make a prediction data set and fill in the predicted probabilities
> TCEPred <- data.frame(PopDensity=1:19)
> TCEPred$Pred <- predict(TCE.lr, newdata=TCEPred, type="response")
> with(TCEPred, addXY(PopDensity, Pred))
> graphics.off()
```

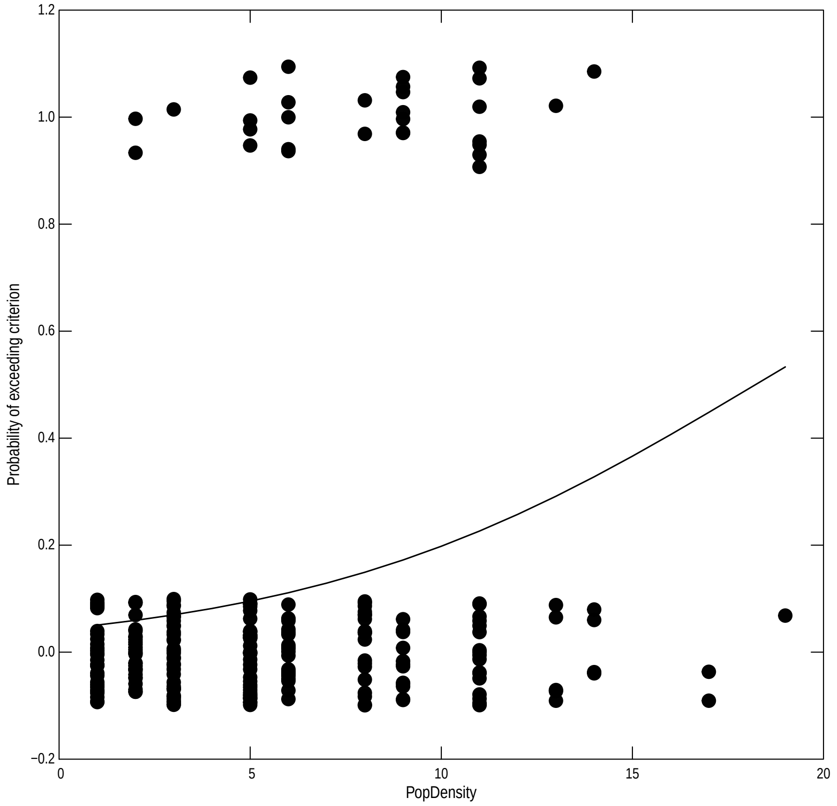

Figure 1. The data and fitted line.

3 Maximum Likelihood Estimation Method

An important first step in any parametric statistical analysis is to plot the data. Figure 2 shows the scatter plots between TCE and PopDensity, Depth, and PctIndLU. None show a strong relation to TCE concentration.

```
> setSweave("graph02", 6 ,6)
> # Create the graphs
> AA.lo <- setLayout(num.cols=2, num.rows=2)
> setGraph(1, AA.lo)
> with(TCEReg, xyPlot(PopDensity, TCE, yaxis.log=TRUE))
> setGraph(2, AA.lo)
> with(TCEReg, xyPlot(Depth, TCE, yaxis.log=TRUE))
> setGraph(3, AA.lo)
> with(TCEReg, xyPlot(PctIndLU, TCE, yaxis.log=TRUE))
> graphics.off()
```

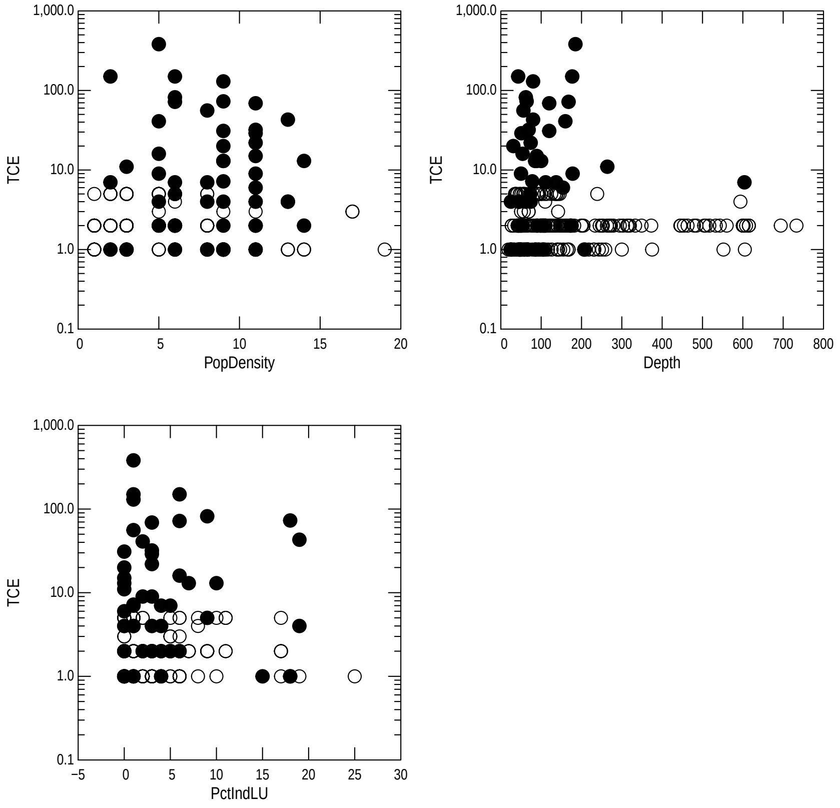

Figure 2. Scatter plots between TCE and PopDensity, Depth, and PctIndLU.

The `censReg` function is used to build a censored-regresion model. The response variable can be numeric, or any of ”lcens,” ”mcens,” or ”qw” class. The explantory variables must be numeric. The code immediately following this paragraph demonstrates is use for the example data. The arguments to censReg are very similar to lm; they are more limited but include an additional argument dist that allows the user to specify the distribution of the residuals and facilitates prediction.

```
> # The censored regression model.
> TCE.cr <- censReg(TCE ~ PopDensity + Depth + PctIndLU, data=TCEReg, dist="lognormal")
> print(TCE.cr)

Call:
censReg(formula = TCE ~ PopDensity + Depth + PctIndLU, data = TCEReg,
    dist = "lognormal")

Coefficients:
             Estimate Std. Error z-score p-value
(Intercept) -2.943870   0.788406 -3.7340  0.0000
PopDensity   0.253026   0.070554  3.5863  0.0003
Depth       -0.004005   0.002297 -1.7439  0.0402
PctIndLU     0.044081   0.052693  0.8366  0.4415

Estimated residual standard error (Unbiased) = 2.838
Distribution: lognormal
Percent standard error: 5608
Positive percent error: 1608
Negative percent error: -94.15

Number of observations = 247, number censored = 194 (78.5 percent)

Loglik(model) = -192 Loglik(intercept only) = -205.5
  Chi-square = 26.95, degrees of freedom = 3, p-value = <0.0001

Computation method: AMLE
```

The printed output is similar to the summary output from a linear regression. The coefficient table list the coefficient, its statndard error, the corresponding z score and the p-value determined from the partial log-likelihood test. Because the response is log-transformed, the residual standard error is expressed as a percentage. The overall significance level is computed from the log-likelihood of the model compared to the log-likelihood of the null model (intercept only). The computation method is ”AMLE,” which produces first-order unbiased estiamtes of the coefficients and standard error. That method also uses a slight variation on the computation of log-likelihood that results in consistent differences in log-likelihood, but the actual value can differ from other methods. A very quick assessment of the fit can be made by looking at the two diagnostic plots.

```
> setSweave("graph03", 6 ,6)
> # Create the graphs
> AA.lo <- setLayout(num.rows=2)
> setGraph(1, AA.lo)
> plot(TCE.cr, which=1, set.up=FALSE)
> setGraph(2, AA.lo)
> plot(TCE.cr, which=2, set.up=FALSE)
> graphics.off()
```

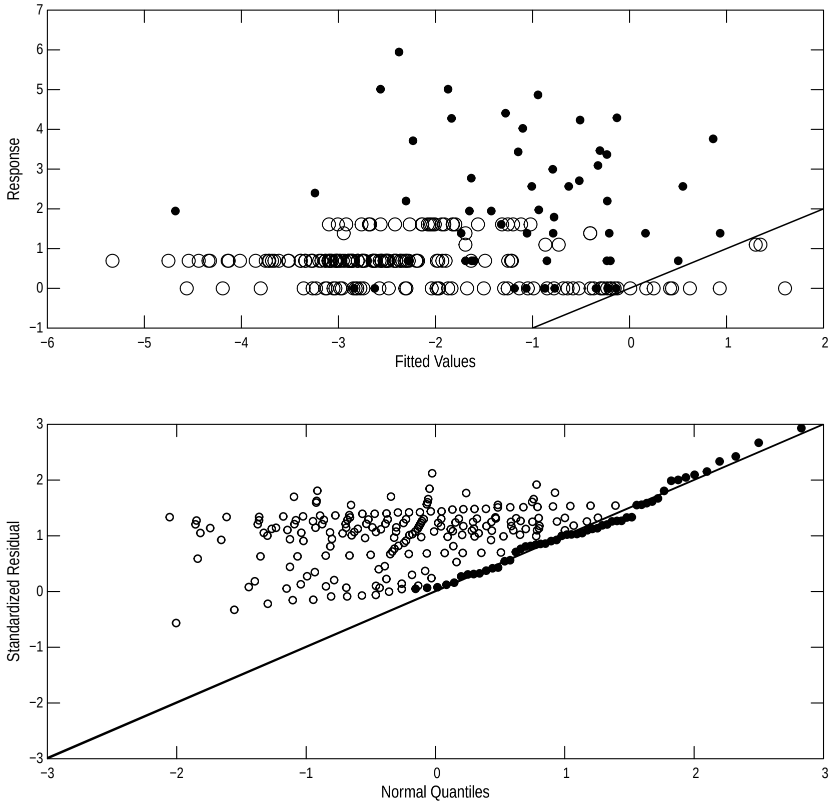

Figure 3. The fitted values and response and q-normal diagnostic plots.

The fitted values and response shows the weakness of the fit, but the q-normal plots indicates no major deviation from normality. For both plots, the uncensored data are shown as solid circles and the censored values as open circles.

The `summary` printout shows more detail about the regression similar to the printed output from the `multReg` function in the `smwrStats` package (Lorenz, 2015). Many of the diagnostic plots from `summary` show working residuals and are designed to portray less heavily censored data than the those in the TCEReg dataset, but the partial residual plots are useful for assessing the linearity of the fit for ech explanatory variable. The code following this paragraph demonstrates the `summary` funciton and shows the printout and generates the partial residual plots for PopDensity and Depth. The partial residual plot for PctIndLU is not shown becuase the p-value is much parger than 0.05.

```
> # The summary output.
> TCE.crsum <- summary(TCE.cr)
> print(TCE.crsum)
```

```
Call:
censReg(formula = TCE ~ PopDensity + Depth + PctIndLU, data = TCEReg,
    dist = "lognormal")

Coefficients:
             Estimate Std. Error z-score p-value
(Intercept) -2.943870   0.788406 -3.7340  0.0000
PopDensity   0.253026   0.070554  3.5863  0.0003
Depth       -0.004005   0.002297 -1.7439  0.0402
PctIndLU     0.044081   0.052693  0.8366  0.4415

Estimated residual standard error (Unbiased) = 2.838
Distribution: lognormal
Percent standard error: 5608
Positive percent error: 1608
Negative percent error: -94.15

Number of observations = 247, number censored = 194 (78.5 percent)

Loglik(model) = -192 Loglik(intercept only) = -205.5
  Chi-square = 26.95, degrees of freedom = 3, p-value = <0.0001

Computation method: AMLE

Pseudo R-squared: 0.1723

  AIC: 394
  BIC: 411.5


Variance inflation factors
PopDensity 1.09
     Depth 1.05
  PctIndLU 1.04

Test criteria
leverage   cooksD
 0.03644  0.84151
       Observations exceeding at least one test criterion
      TCE  ycen    yhat  resids leverage    cooksD
7   0.0000  TRUE -0.8491 -1.7530  0.10497 1.250e-02
55  0.0000  TRUE -0.9887 -1.6747  0.04355 4.145e-03
56  1.0986  TRUE  1.3497 -2.4266  0.04123 8.198e-03
62  0.0000 FALSE -0.2176  0.2176  0.04768 7.727e-05
67  1.0986  TRUE  1.3016 -2.3952  0.04125 7.991e-03
68  3.7612 FALSE  0.8626  2.8986  0.06019 1.777e-02
70  0.0000  TRUE  1.6034 -3.3742  0.05733 2.280e-02
92  0.0000 FALSE -0.1375  0.1375  0.04819 3.122e-05
93  4.2905 FALSE -0.1295  4.4200  0.04825 3.230e-02
94  0.0000  TRUE  0.9307 -2.8882  0.06042 1.772e-02
95  1.3863 FALSE  0.9347  0.4516  0.06044 4.334e-04
```

```
130 1.9459 FALSE -4.6804  6.6263  0.03834 5.651e-02
145 0.6931  TRUE -3.7124 -0.3611  0.04169 1.837e-04
149 0.6931  TRUE -3.0140 -0.5334  0.04520 4.379e-04
150 0.6931  TRUE -3.6603 -0.3724  0.03957 1.846e-04
152 0.6931  TRUE -4.5436 -0.2132  0.03809 5.811e-05
162 0.6931  TRUE -2.8538 -0.5797  0.03887 4.390e-04
169 0.6931  TRUE -3.6844 -0.3671  0.04054 1.843e-04
173 0.6931  TRUE -5.3293 -0.1212  0.06117 3.163e-05
175 0.6931  TRUE -3.3905 -0.4347  0.06234 4.159e-04
178 0.6931  TRUE -4.1417 -0.2775  0.04314 1.127e-04
179 0.6931  TRUE -2.7476 -0.6119  0.05666 7.399e-04
180 0.6931  TRUE -2.8717 -0.5744  0.05989 6.941e-04
206 0.0000  TRUE -4.5637 -0.3284  0.03885 1.408e-04
227 0.6931  TRUE -1.2498 -1.1891  0.04516 2.174e-03
242 1.6094  TRUE -1.1176 -0.8579  0.04658 1.171e-03
243 1.3863  TRUE -2.9459 -0.3770  0.04552 2.205e-04
246 0.6931  TRUE -4.7526 -0.1847  0.03775 4.317e-05
```

```
> setSweave("graph04", 6 ,6)
> # Create the graphs
> AA.lo <- setLayout(num.rows=2)
> setGraph(1, AA.lo)
> plot(TCE.crsum, which="PopDensity", set.up=FALSE)
> setGraph(2, AA.lo)
> plot(TCE.crsum, which="Depth", set.up=FALSE)
> graphics.off()
```

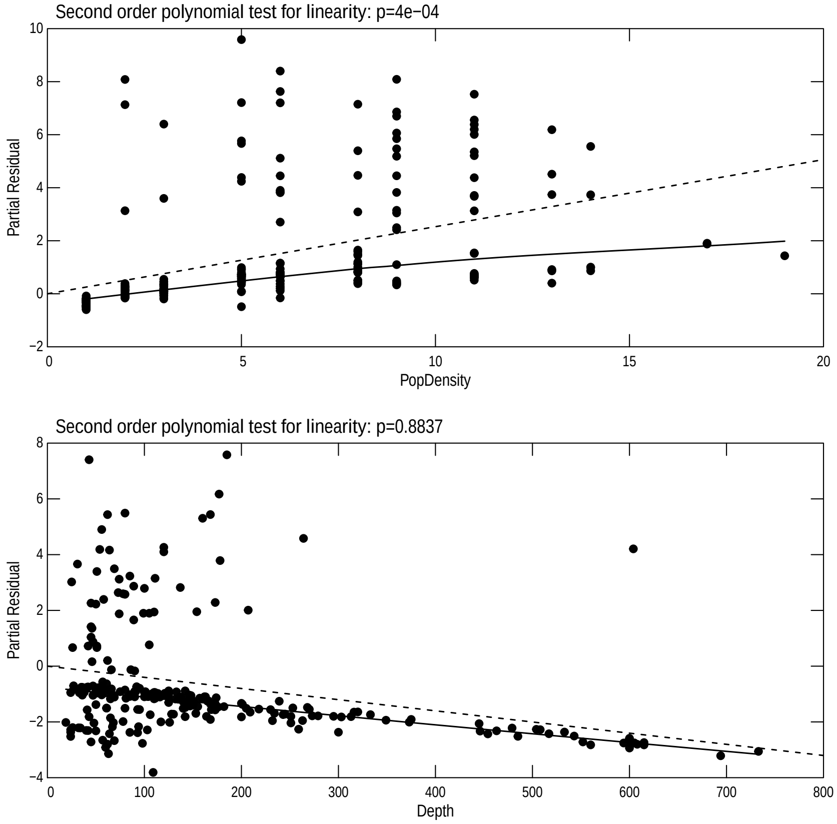

Figure 4. The partial residual plots for PopDensity and Depth.

The printed output from `summary` show many observations that exceed at least one of the test criteria, leverage or Cook's D (Helsel and Hirsch, 2002). For these data, all of the exceedences are for leverage, indicating some values are far from the bulk of the data. The partial residual plot for PoPDensity indicates pretty substantial nonlinearity, but the plot for Depth does not. The code below constructs the revised model, dropping PctIndLU and using a log-transform for PopDensity. The `summary` report is shown and the diagnostics plots from the model and the partial residual plots are shown from the `summary` output. The AIC and BIC are both much smaller for the revised model, indicating a much better overall fit. Figures 5 and 6 also indicate a better fit for the revised model than the original model.

```
> # The revised censored regression model.
> TCE.cr <- censReg(TCE ~ log(PopDensity) + Depth, data=TCEReg, dist="lognormal")
> # The summary output.
> TCE.crsum <- summary(TCE.cr)
> print(TCE.crsum)

Call:
censReg(formula = TCE ~ log(PopDensity) + Depth, data = TCEReg,
    dist = "lognormal")

Coefficients:
                Estimate Std. Error z-score p-value
(Intercept)     -3.97145   1.003544  -3.957  0.0000
log(PopDensity)  1.72719   0.423271   4.081  0.0000
Depth           -0.00397   0.002207  -1.799  0.0381

Estimated residual standard error (Unbiased) = 2.768
Distribution: lognormal
Percent standard error: 4616
Positive percent error: 1493
Negative percent error: -93.72

Number of observations = 247, number censored = 194 (78.5 percent)

Loglik(model) = -188.3 Loglik(intercept only) = -205.5
  Chi-square = 34.41, degrees of freedom = 2, p-value = <0.0001

Computation method: AMLE

Pseudo R-squared: 0.2363

  AIC: 384.5
  BIC: 398.6


Variance inflation factors
log(PopDensity) 1.01
          Depth 1.01

Test criteria
leverage   cooksD
 0.02429  0.79086
        Observations exceeding at least one test criterion
       TCE  ycen   yhat   resids leverage    cooksD
115 0.6931  TRUE -3.244 -0.43547  0.02649 2.305e-04
121 0.0000  TRUE -3.383 -0.58870  0.03107 4.988e-04
123 0.6931  TRUE -4.791 -0.15903  0.02693 3.128e-05
125 0.6931  TRUE -4.775 -0.16090  0.02647 3.145e-05
130 1.9459 FALSE -5.172  7.11817  0.03953 9.442e-02
145 0.6931  TRUE -3.318 -0.41731  0.04161 3.431e-04
149 0.6931  TRUE -2.762 -0.56703  0.04203 6.403e-04
150 0.6931  TRUE -3.267 -0.42989  0.03955 3.446e-04
152 0.6931  TRUE -4.456 -0.20223  0.03714 7.125e-05
162 0.6931  TRUE -2.603 -0.61560  0.03600 6.385e-04
169 0.6931  TRUE -3.291 -0.42405  0.04049 3.439e-04
173 0.6931  TRUE -5.684 -0.07861  0.06153 1.877e-05
175 0.6931  TRUE -3.135 -0.46316  0.05838 6.142e-04
178 0.6931  TRUE -3.633 -0.34616  0.04040 2.286e-04
187 0.6931  TRUE -4.890 -0.14774  0.02990 3.016e-05
```

```
206 0.0000  TRUE -4.476 -0.31565  0.03789 1.774e-04
239 0.6931  TRUE -3.347 -0.41035  0.02985 2.323e-04
243 1.3863  TRUE -2.738 -0.39071  0.04109 2.967e-04
246 0.6931  TRUE -5.156 -0.12059  0.03894 2.666e-05
```

```
> # Overall and Q-normal plots
> setSweave("graph05", 6 ,6)
> # Create the graphs
> AA.lo <- setLayout(num.rows=2)
> setGraph(1, AA.lo)
> plot(TCE.cr, which=1, set.up=FALSE)
> setGraph(2, AA.lo)
> plot(TCE.cr, which=2, set.up=FALSE)
> graphics.off()
> setSweave("graph06", 6 ,6)
> # Parial residual plots
> AA.lo <- setLayout(num.rows=2)
> setGraph(1, AA.lo)
> plot(TCE.crsum, which="log(PopDensity)", set.up=FALSE)
> setGraph(2, AA.lo)
> plot(TCE.crsum, which="Depth", set.up=FALSE)
> graphics.off()
```

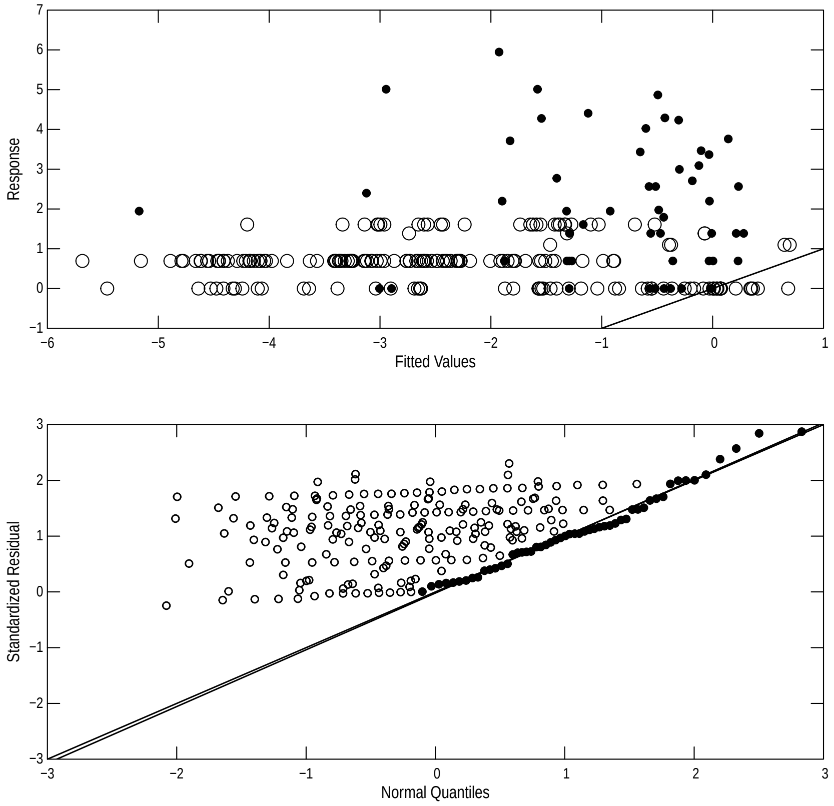

Figure 5. The over all fit and q-normal diagnostic plots for the revised model.

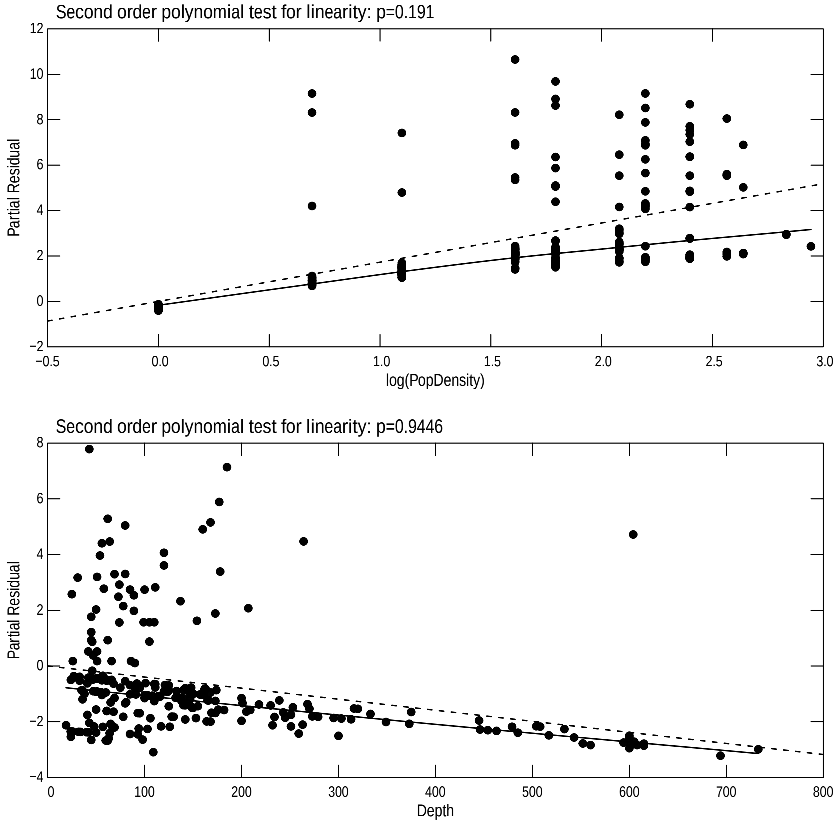

Figure 6. The partial residual plots for PopDensity and Depth for the revised model.

References

[1] Harrel, F.E., 2001, Regression modeling strategies with applications to linear models, logistic regression, and survival analysis: New York, Springer, 568 p.

[2] Helsel, D.R. 2012, Statistics for Censored Environmental Data Using Minitab and R: New York, Wiley, 324 p.

[3] Helsel, D.R., and Hirsch, R.M., 2002, Statistical methods in water resources: U.S. Geological Survey Techniques of Water-Resources Investigations, book 4, chap. A3, 522 p.

[4] Lorenz, D.L., 2015, smwrStats–An R package for the analysis of hydrologic data, Version 0.7.3: U.S. Geological Survey Open File Report, ? p.

[5] McCullagh, P, and Nedler, J.A., 1999, Generalized linear models: Boca Raton, Fla., Chappman and Hall/CRC, 511 p.