

Trend Analysis of Uncensored Major Ions

Dave Lorenz

July 27, 2017

Abstract

This example illustrates the data manipulations for the seasonal Kendall analysis of uncensored data. Major ions are typically uncensored in natural waters and provide a useful example. This example also uses a common time frame for all of the trend tests. The common time frame facilitates comparing trends among the stations. Most often users will want to divide trend analyses into similar groups of analytes like major ions, nutrients and so forth because they will be analyzed in similar ways and will have common sampling time frames.

Contents

1	Introduction	2
2	Summarize the Sample Data	3
3	Set up the Project	6
4	Flow Adjustment	8
5	Seasonal Kendall Trend Test	13
6	Trend Results	15
7	Further Remarks	19

1 Introduction

The data used in this application are a small subset of the data used by Schertz and others (1991). The data are samples taken from water year 1969 (October, 1968) through water year 1989 (September, 1989). Nineteen stations were selected and only calcium and chloride were selected for the major ions. The data were modified by removing the remark columns associated with those constituents to make the analysis more straightforward.

```
> # Load the restrend and smwrBase packages and the data
> library(restrend)
> library(smwrBase)
> data(EstrendSub)
> head(EstrendSub)
```

```
      STAID      DATES      QI QD RN.organic PN.organic RAMmonia
1 07227500 1968-10-01   7.6 NA          NA          NA
2 07227500 1968-10-03   5.3 NA          NA          NA
3 07227500 1968-10-16 532.0 NA          NA          NA
4 07227500 1968-10-19  17.0 NA          NA          NA
5 07227500 1968-11-01  17.0 NA          NA          NA
6 07227500 1968-12-01   6.6 NA          NA          NA
      PAmmonia RKjeldahl PKjeldahl RTotal.P PTotal.P RCopper PCopper
1          NA          NA          NA          NA          NA
2          NA          NA          NA          NA          NA
3          NA          NA          NA          NA          NA
4          NA          NA          NA          NA          NA
5          NA          NA          NA          NA          NA
6          NA          NA          NA          NA          NA
      RIron PIron Calcium Chloride
1          NA      95      280
2          NA      NA      106
3          NA      42      106
4          NA     121     435
5          NA     150     512
6          NA     138     510
```

2 Summarize the Sample Data

In general, it is desirable, but not necessary, to subset the data before proceeding with the analysis of a subset of the constituents. Before these data are subsetted, the `FLOW` column must be created. The flow data are in two columns `QI`, the flow at the time of the sample; and `QD`, the mean flow on the day of the sample. The `coalesce` function in the `smwrBase` package can be used to select the non-missing value for flow.

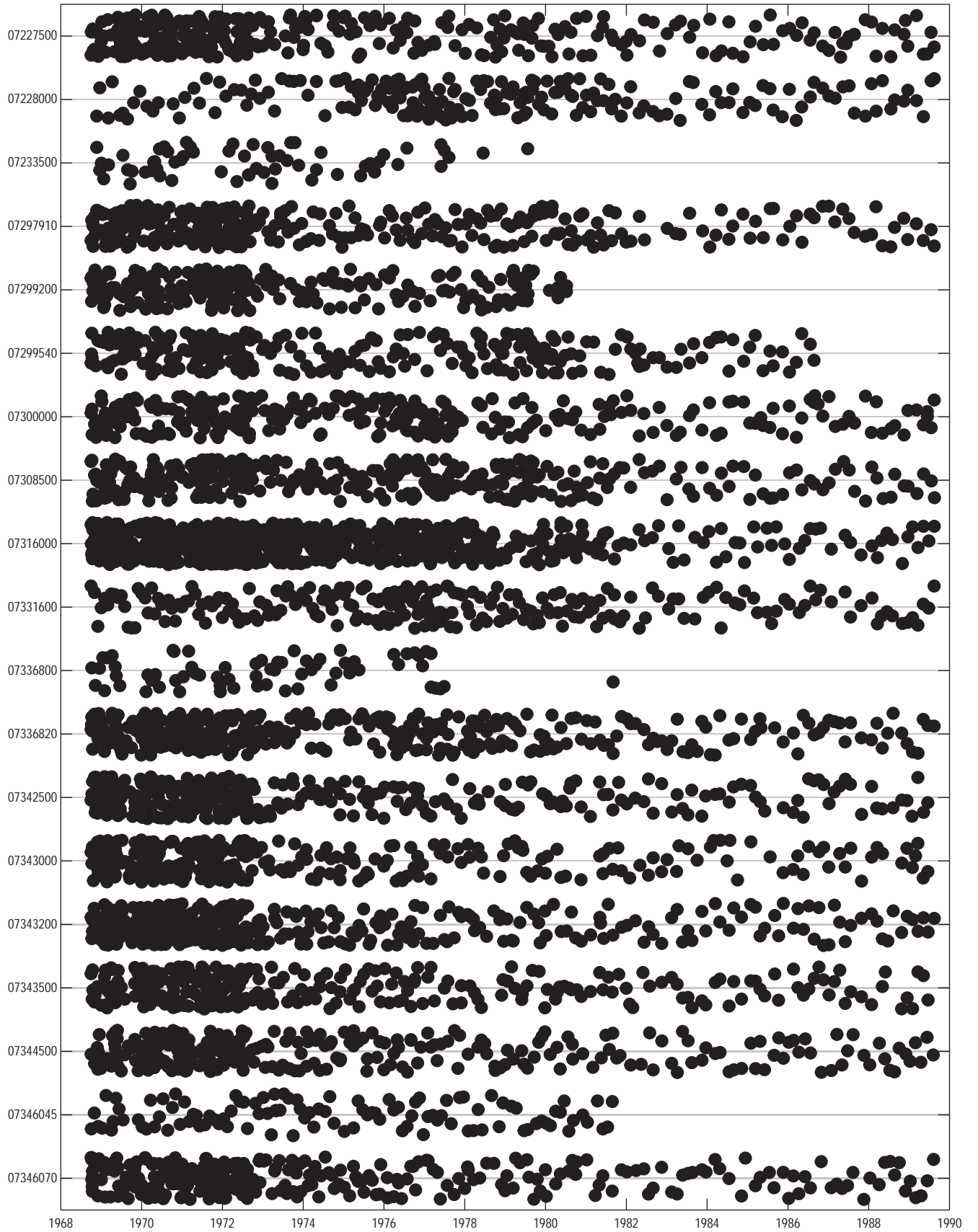
```
> # Compute FLOW, the coalesce function is in smwrBase
> EstrendSub <- transform(EstrendSub, FLOW=coalesce(QI, QD))
> # Create the subset
> Majors <- subset(EstrendSub, select=c("STAID", "DATES", "FLOW",
+   "Calcium", "Chloride"))
```

The `sampReport` function creates a simple PDF file that contains a report of the sample date ranges and graph of samples for each site. It can be used to help define the starting and ending date ranges for the trend tests as well as identifying sample gaps and other sampling issues.

```
> # Create the report
> sampReport(Majors, DATES="DATES", STAID="STAID", file="MajorIonSampling")
```

The call to `sampReport` returns the file name invisibly (`MajorIonSampling.pdf`). Because it is a full-size portrait PDF file, it is inserted here with compressed pages. The report gives the actual begin and end dates for sampling and the graph shows the sampling dates for each station. It is easy to see that 5 stations (07233500, 07299200, 07299540, 07336800, and 07346045) were not sampled for the entire retrieval period.

	STAIID	FirstSamp	LastSamp	NumSamp
1	07227500	1968-10-01	1989-08-15	350
2	07228000	1968-11-21	1989-08-16	198
3	07233500	1968-11-21	1979-07-24	62
4	07297910	1968-10-01	1989-08-15	308
5	07299200	1968-10-01	1980-07-09	217
6	07299540	1968-10-01	1986-08-25	276
7	07300000	1968-10-01	1989-08-16	283
8	07308500	1968-10-01	1989-08-18	336
9	07316000	1968-10-01	1989-08-16	694
10	07331600	1968-10-01	1989-08-15	228
11	07336800	1968-10-07	1981-09-03	69
12	07336820	1968-10-01	1989-08-15	393
13	07342500	1968-10-01	1989-06-19	325
14	07343000	1968-10-01	1989-06-19	317
15	07343200	1968-10-01	1989-08-16	366
16	07343500	1968-10-01	1989-06-21	303
17	07344500	1968-10-01	1989-08-07	294
18	07346045	1968-10-03	1981-08-27	109
19	07346070	1968-10-01	1989-08-10	300



3 Set up the Project

The user must balance the need to include as many stations as possible and the targeted time frame for the trend estimation. For these data, 5 stations have incomplete record, but to include all of those stations, the analysis period would need to be much shorter, though water year 1978. This example will use the full retrieval period.

The `(setProj)` function sets up the trend estimation project. There are many arguments to `(setProj)`, see the documentation for details. The constituent names or response variable names are referred to as `Snames` in keeping with the names used in the original ESTREND.

After projects have been set up, the user can get a list of the projects by using `lsProj` or can specify a project to use with `useProj`. The function `useProj` must be used to continue working on a project after the user quits from the R session.

```
> # Set up the project
> setProj("majors", Majors, STAID="STAID", DATES="DATES",
+        Snames=c("Calcium", "Chloride"), FLOW="FLOW",
+        type="seasonal", Start="1968-10-01", End="1989-10-01")

[1] "majors"
```

The `(setProj)` function creates a folder in the users workspace with that name. That folder contains R data that are updated after each successful call to an analysis function in `restrend`. Table 1 describes the data created in this example's call to `(setProj)`. Any object of class "matrix" or "by" are indexed by station and sname.

Table 1. The data created by `(setProj)`.

Name	Class	Description
<code>estrend.cl</code>	list	A record of the calls to analysis functions.
<code>estrend.cn</code>	matrix	A description of the censoring. May be "none," "left," or "multiple."
<code>estrend.cp</code>	matrix	The percent of observations that are left-censored.
<code>estrend.df</code>	by	The dataset, contains STAID, DATES, FLOW, and the response variable.
<code>estrend.in</code>	list	Information about the project, such as the start and end dates and the names of columns in each dataset.
<code>estrend.sl</code>	by	Details from the seasonal selection process. Each is a list from the potential comparisons from 12, 6, 4, and 3 seasons per year definition. See Lorenz (2014) for details.
<code>estrend.ss</code>	matrix	The "best" seasonal definition from the analysis recorded in <code>estrend.sl</code> .
<code>estrend.st</code>	matrix	The status for each station and sname. Must be "OK" to continue with the trend analysis.

It is useful to verify which stations and snames will be analyzed and what the seasonal definitions are. The user need only enter the name of the R data object in the console. For these data, the seasonal definition is 0 in all cases where the status is not "OK."

```
> # Which are OK?
> estrend.st
```

```

          snames
stations  Calcium      Chloride
07227500 "OK"           "OK"
07228000 "OK"           "OK"
07233500 "short record" "short record"
07297910 "OK"           "OK"
07299200 "short record" "short record"
07299540 "short record" "short record"
07300000 "OK"           "OK"
07308500 "OK"           "OK"
07316000 "OK"           "OK"
07331600 "OK"           "OK"
07336800 "short record" "short record"
07336820 "OK"           "OK"
07342500 "OK"           "OK"
07343000 "OK"           "OK"
07343200 "OK"           "OK"
07343500 "OK"           "OK"
07344500 "OK"           "OK"
07346045 "short record" "short record"
07346070 "OK"           "OK"

```

```

> # What seasonal definition?
> estrend.ss

```

```

          snames
stations  Calcium Chloride
07227500      6        6
07228000      6        6
07233500      0        0
07297910      6        6
07299200      0        0
07299540      0        0
07300000      6        6
07308500      6        6
07316000     12       12
07331600     12       12
07336800      0        0
07336820     12       12
07342500     12       12
07343000     12       12
07343200     12       12
07343500     12       12
07344500     12       12
07346045      0        0
07346070     12       12

```

4 Flow Adjustment

Computing flow-adjusted concentrations (flow adjustment) is an optional step in seasonal Kendall trend analysis. It is only appropriate for uncensored or slightly censored data. If the data are censored, the censoring is ignored and the values are taken as the detection limit. Flow adjustment is performed using the `flowAdjust` function and can immediately follow the call to `setProj`.

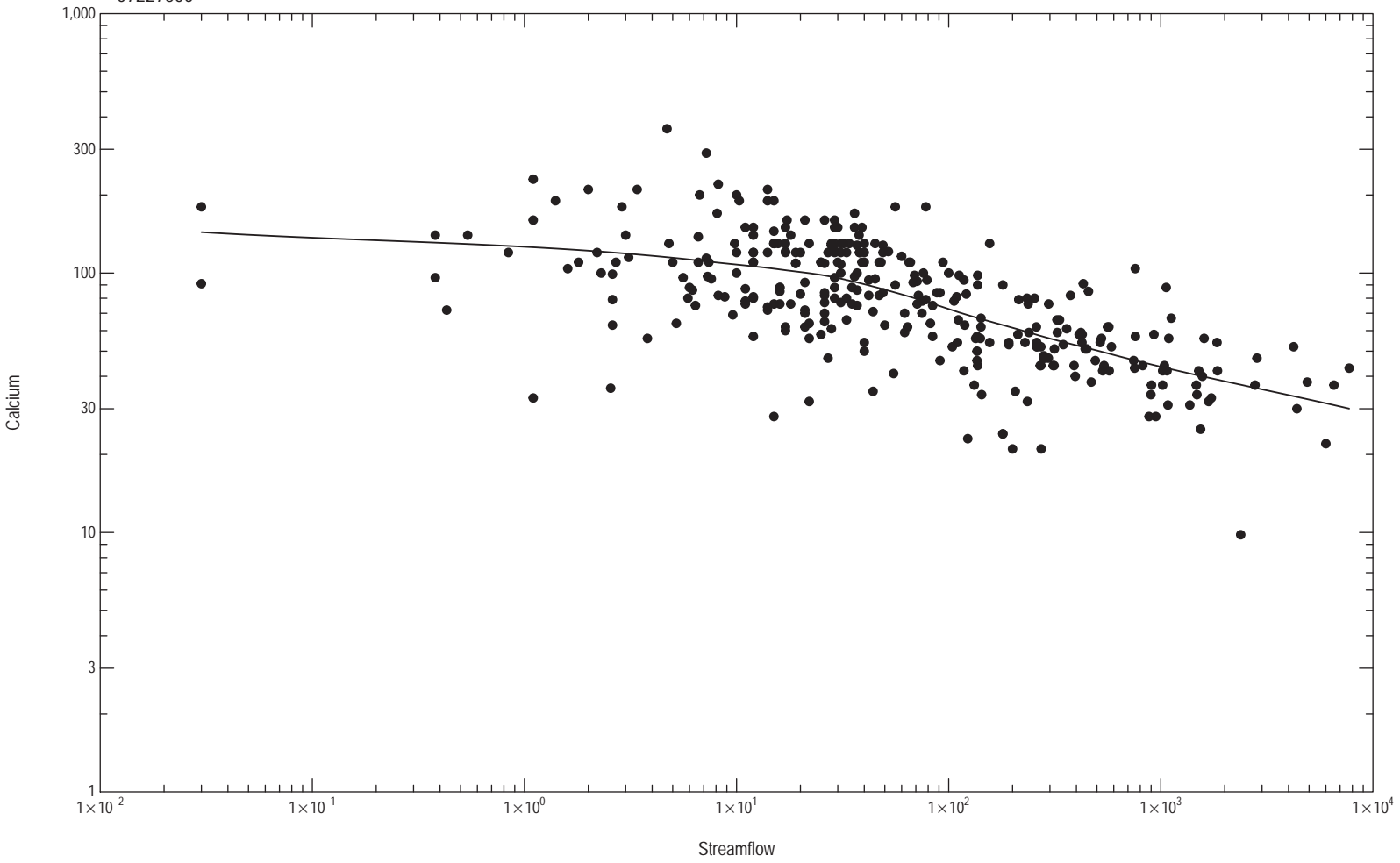
By default, all stations and snames are flow adjusted by `flowAdjust`. But specific combinations can be separately adjusted, using a different span for the LOWESS procedure for example, or if no satisfactory fit can be found, selected combinations can be completely undone by the `undoFA` function. Note that no relation between flow and concentration is not necessarily an unsatisfactory fit.

The `flowAdjust` function creates a PDF report, and returns the name of the report. The report shows graphs of flow and concentration by station on each page. Up to 6 combinations are shown on a page. For any seasonal Kendall trend test with flow adjustment, the user should review all flow-concentration relation. Only 2 pages are shown in this example, the first illustrates an acceptable fit and the second a marginal fit. The user may choose to revise other flow adjustments or accept all flow adjustments, but those for station 07331600 were selected to demonstrate customized flow adjustment.

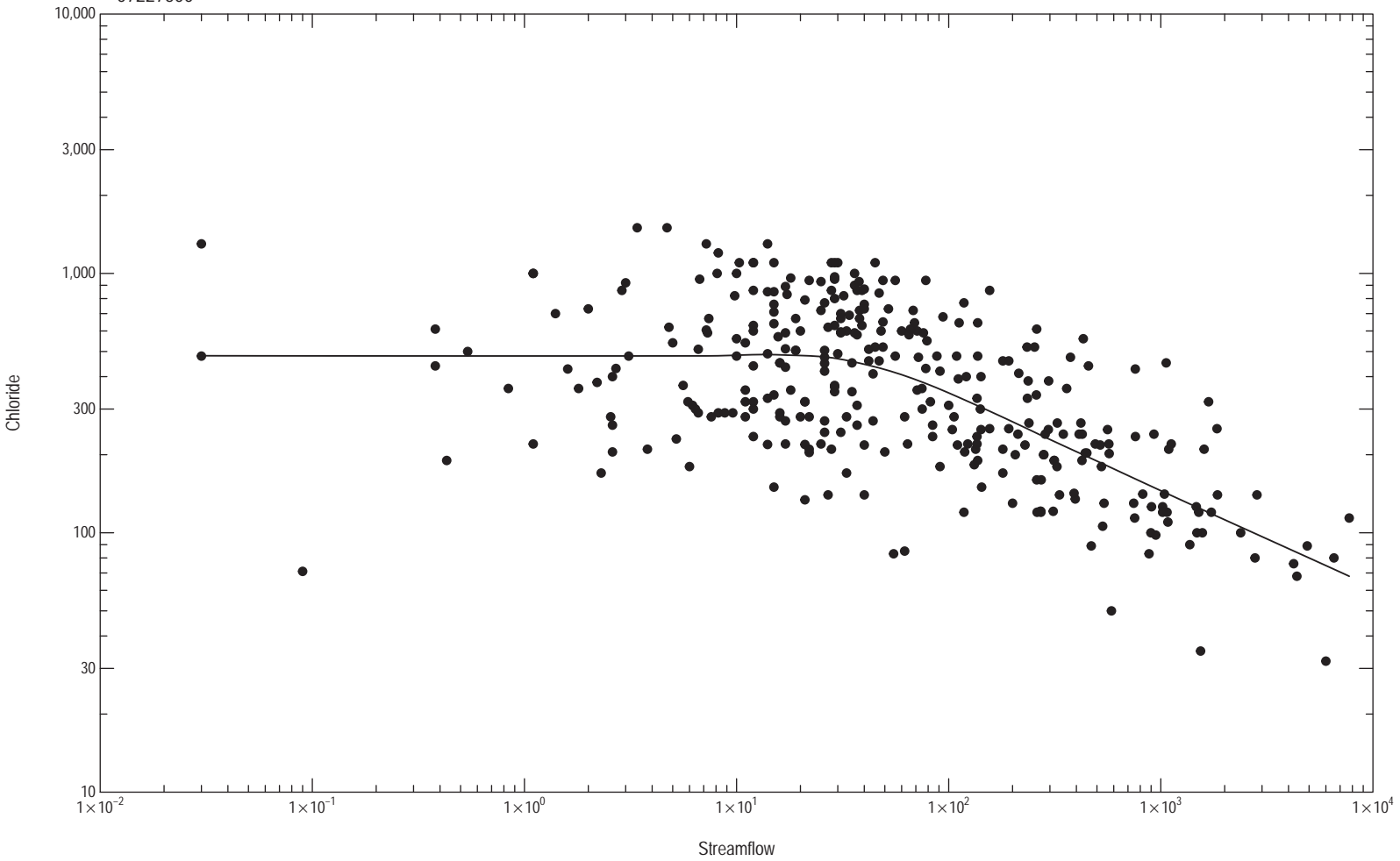
```
> # Do the flow adjustment accepting all defaults
> flowAdjust()
```

```
[1] "majors_fa.pdf"
```

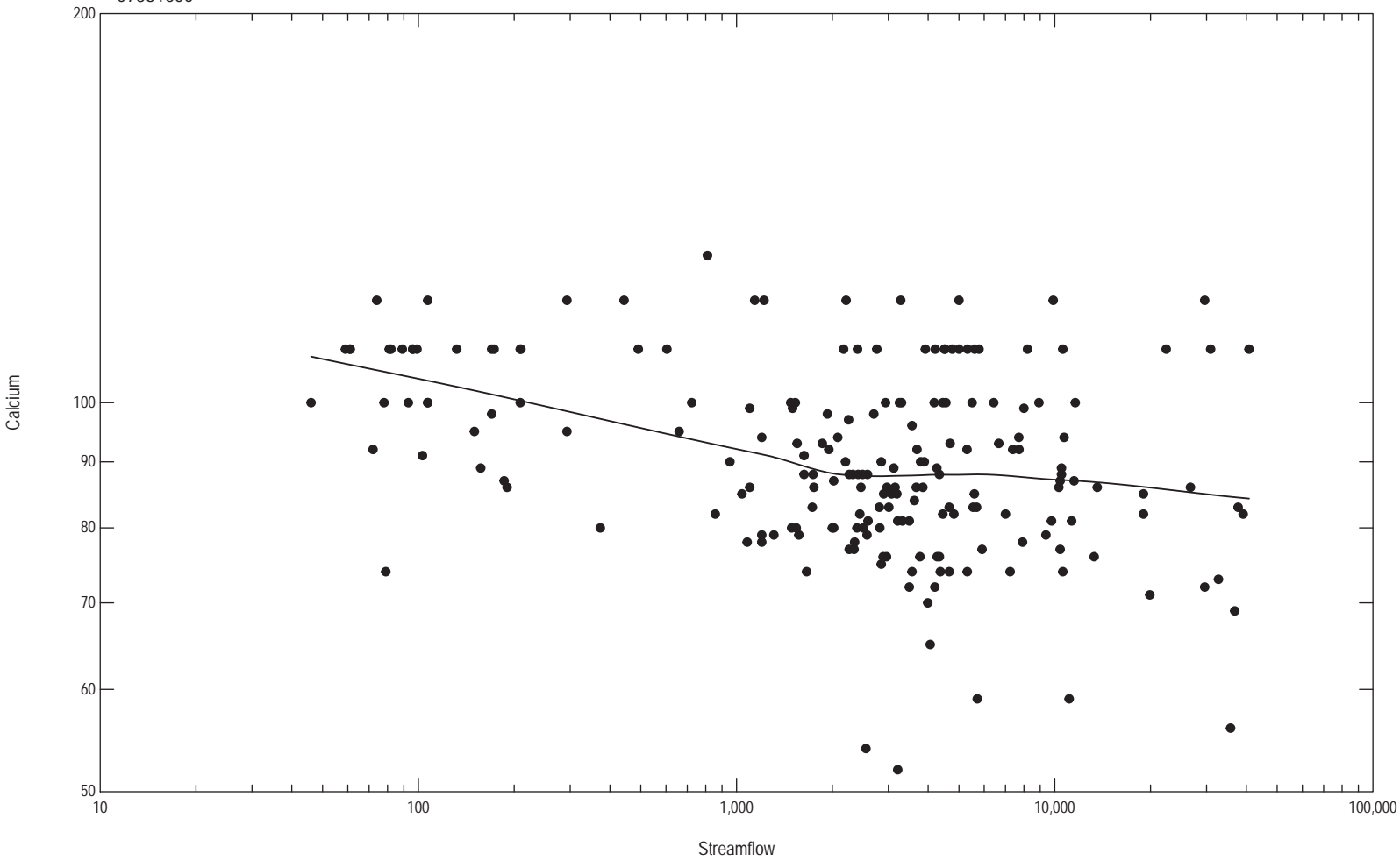

07227500



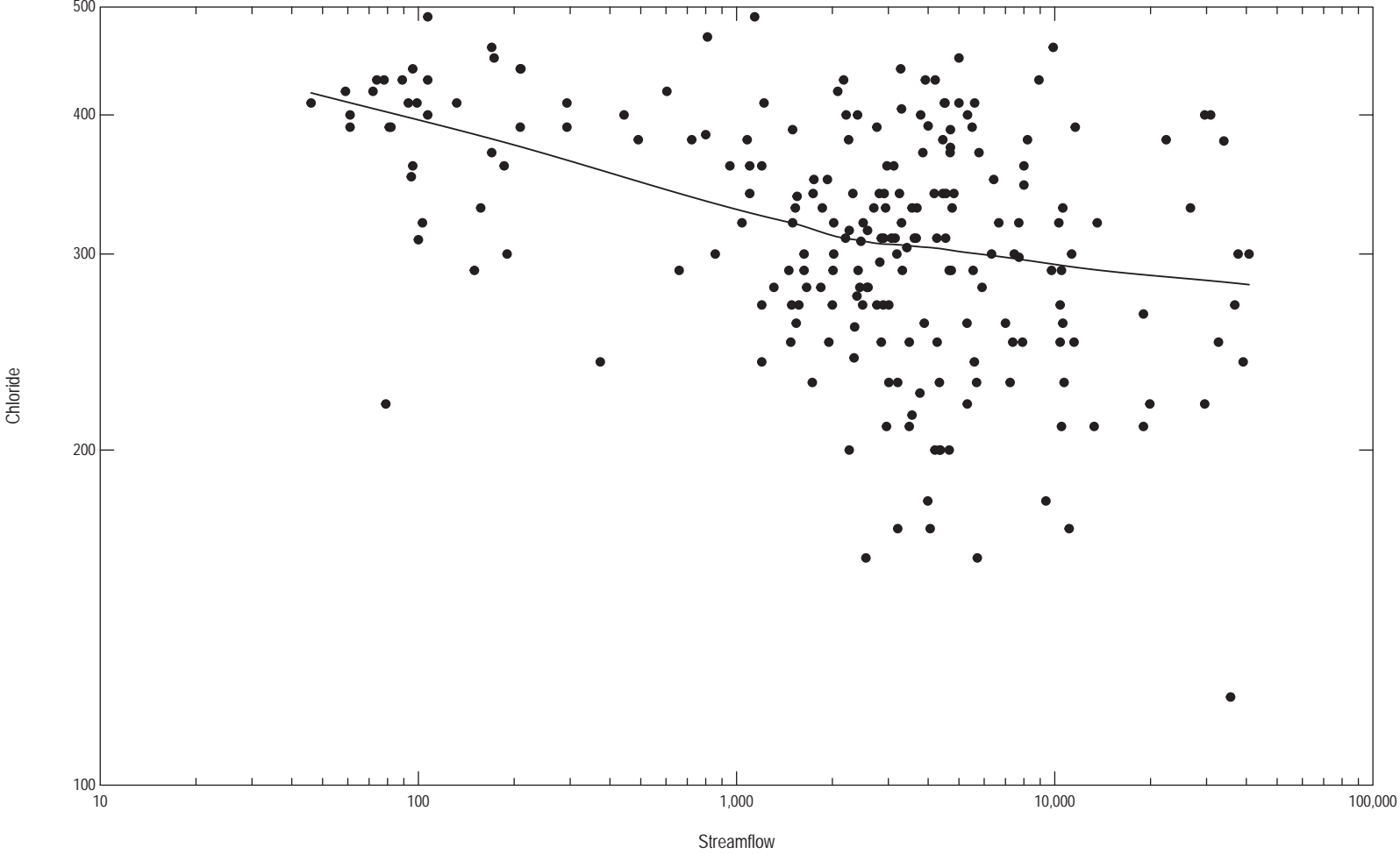
07227500



07331600



07331600

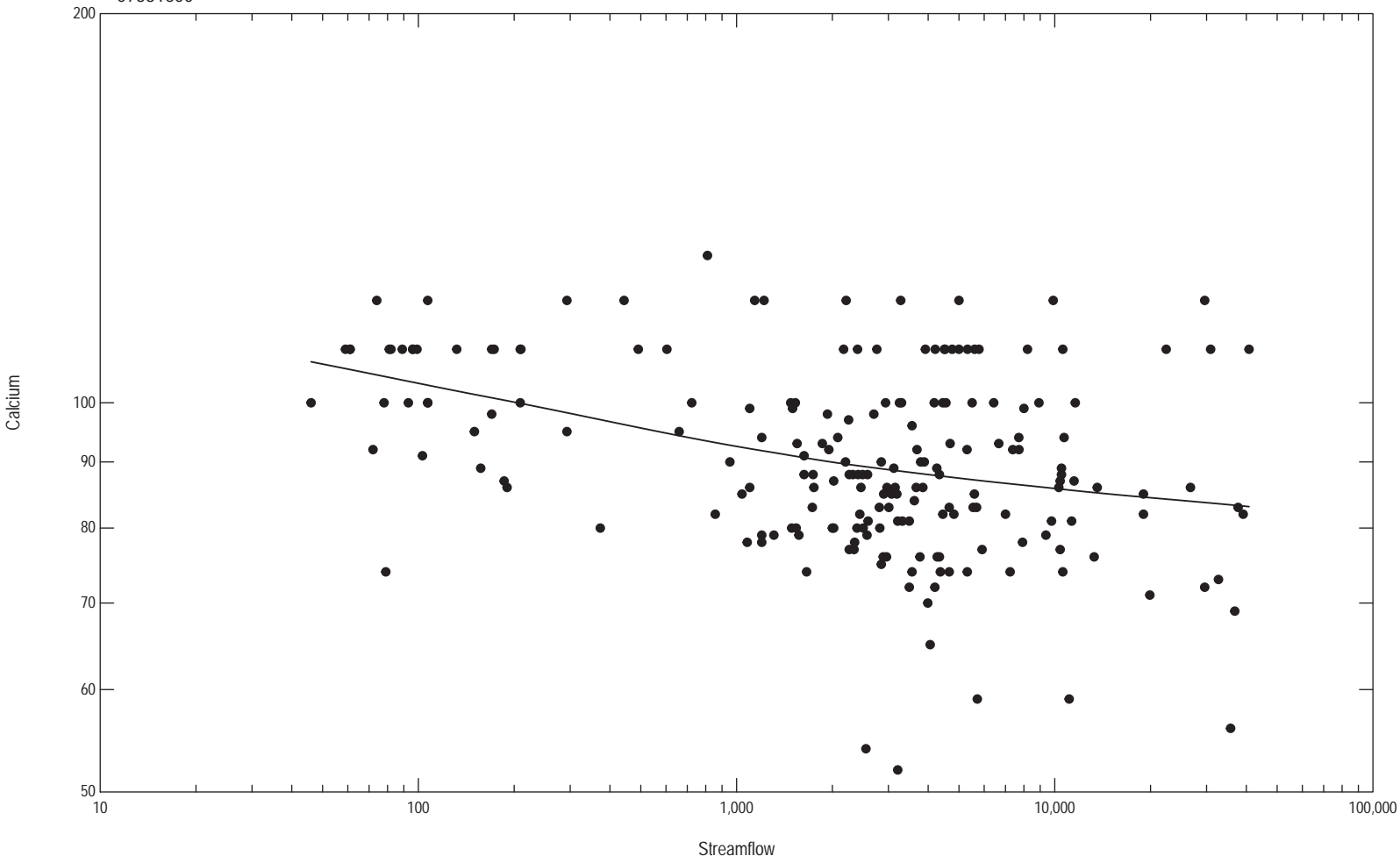


The revised fit shows an improved, more smooth fit to the data.

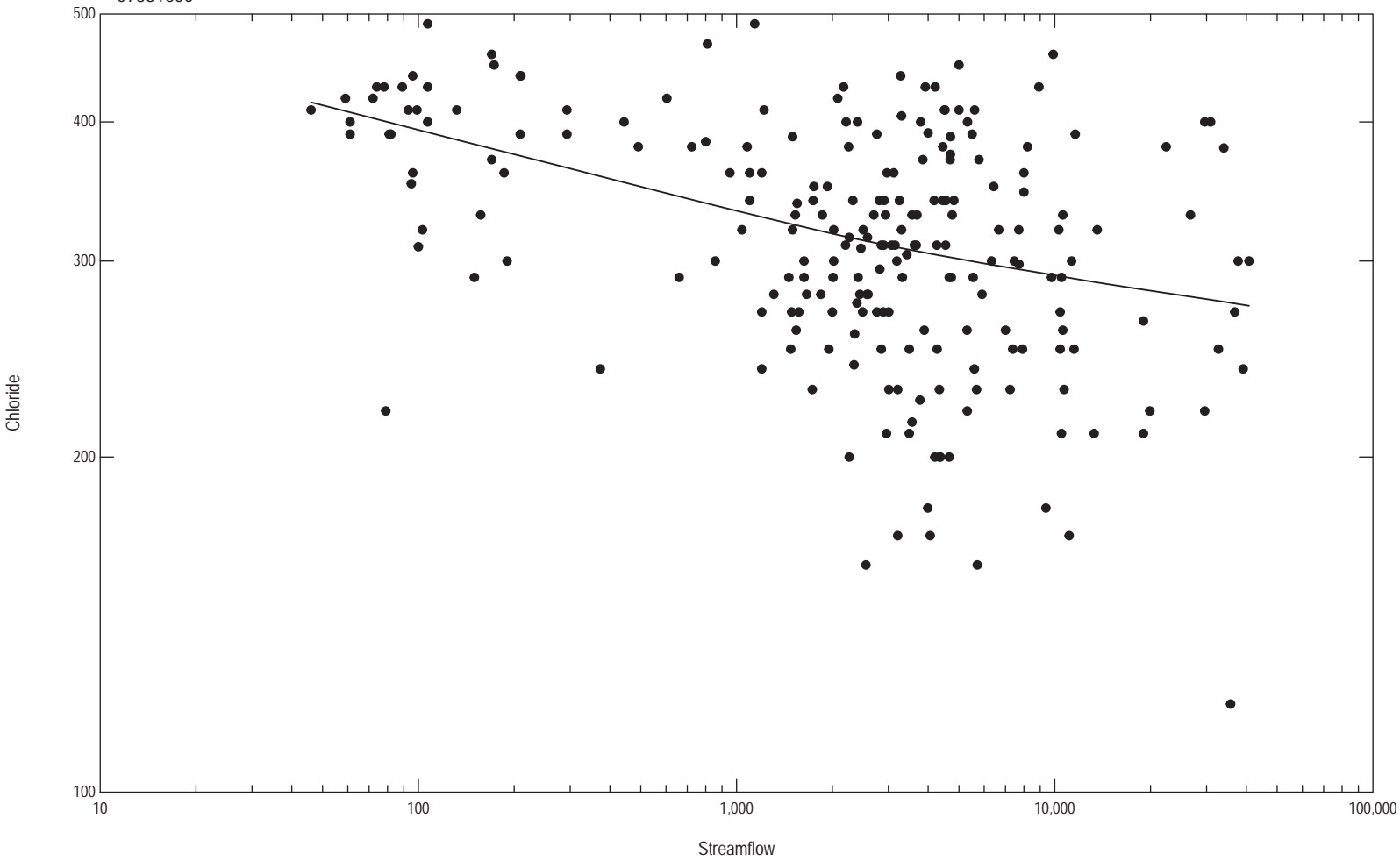
```
> # Do the flow adjustment accepting all defaults  
> flowAdjust(Station="07331600", Snames=c("Calcium", "Chloride"), span=1)
```

```
[1] "majors_fa_01.pdf"
```

07331600



07331600



5 Seasonal Kendall Trend Test

After the optional flow-adjustment, these data are ready for the seasonal Kendall trend test. The function `SKTrends` executes the trend test on all valid combinations of stations and snames. It can also execute the test on subsets if some changes need to be made. An important argument is `nseas`, which can be used to force all analyses to use the same seasonal definition. This is essential for the regional seasonal Kendall test and an important consideration for other regional assessments because it levels the playing field for determining significant trends.

The `SKTrends` function also creates a PDF file that contains the result of the analysis and a series graph on each page. See the documentation for `seriesPlot` for information about that graph. The file reports the results for each sname by station with the flow-adjusted results following the untransformed results. Most trends are very small for these data; only the reports for Calcium at 07228000 is shown.

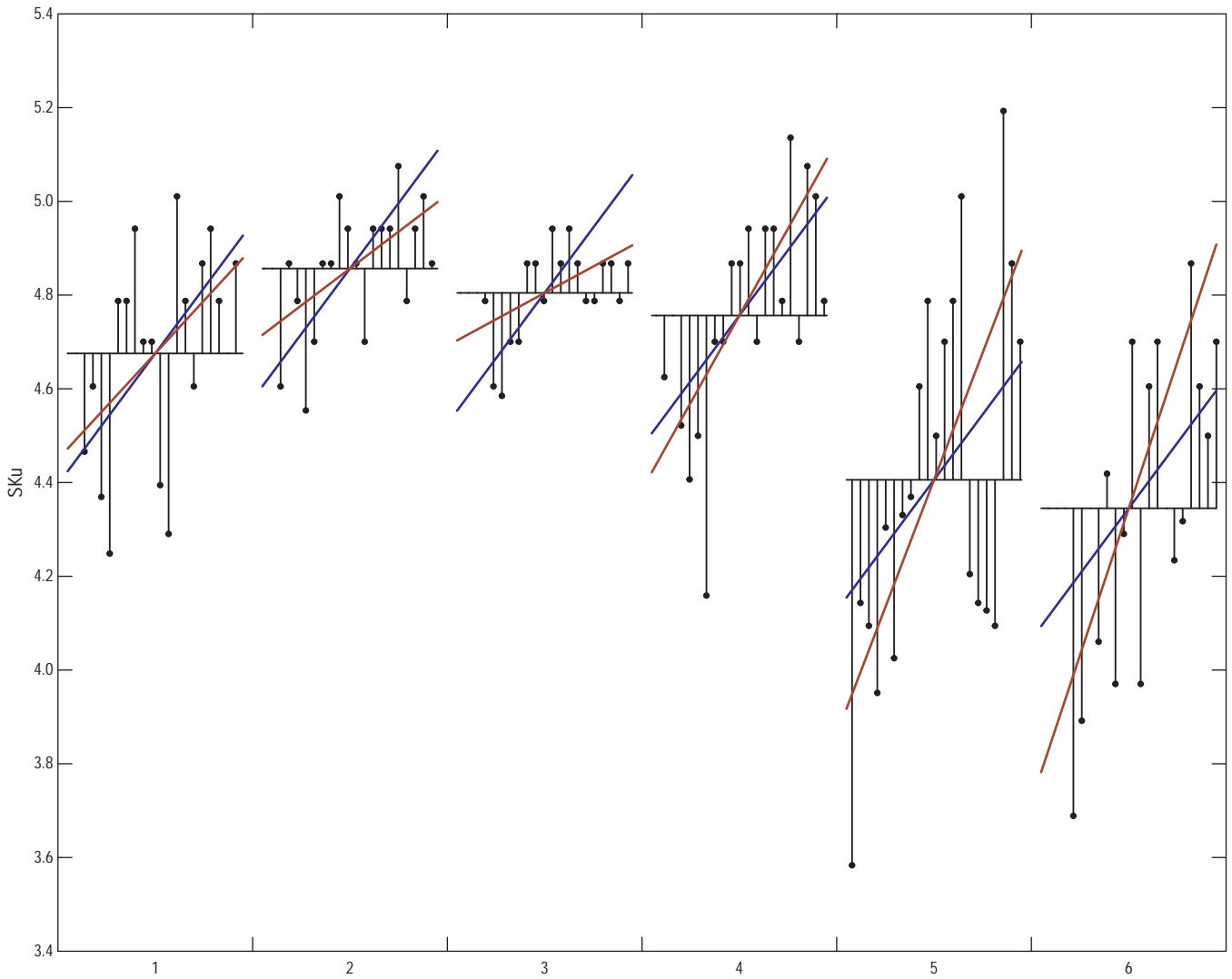
```
> # Trend tests, accepting default seasons
> SKTrends()
```

```
[1] "majors_sk.pdf"
```

07228000 Calcium

Seasonal Kendall with correlation correction

data: log(Calcium) (21 years and 6 seasons)
tau = 0.41513, p-value = 0.0003365
alternative hypothesis: true slope is not equal to 0
sample estimates:
slope median.data median.time
0.02154697 4.70048046 10.50000000



6 Trend Results

When completed, or to check on intermediate results, the estimated trends can be extracted using the `getTrends` function. By default, all stations and snames are extracted. The output dataset is explained in the documentation for `getTrends`. The user has the option to set a significance level to determine whether there is a significant trend, the default level is 0.05.

```
> # get the trends
> majors.tnd <- getTrends()
> print(majors.tnd)
```

	Station	Response		Type	NumYears
1	07227500	Calcium	uncensored	seasonal Kendall	21
2	07227500	Calcium	flow-adjusted	seasonal Kendall	21
3	07227500	Chloride	uncensored	seasonal Kendall	21
4	07227500	Chloride	flow-adjusted	seasonal Kendall	21
5	07228000	Calcium	uncensored	seasonal Kendall	21
6	07228000	Calcium	flow-adjusted	seasonal Kendall	21
7	07228000	Chloride	uncensored	seasonal Kendall	21
8	07228000	Chloride	flow-adjusted	seasonal Kendall	21
9	07297910	Calcium	uncensored	seasonal Kendall	21
10	07297910	Calcium	flow-adjusted	seasonal Kendall	21
11	07297910	Chloride	uncensored	seasonal Kendall	21
12	07297910	Chloride	flow-adjusted	seasonal Kendall	21
13	07300000	Calcium	uncensored	seasonal Kendall	21
14	07300000	Calcium	flow-adjusted	seasonal Kendall	21
15	07300000	Chloride	uncensored	seasonal Kendall	21
16	07300000	Chloride	flow-adjusted	seasonal Kendall	21
17	07308500	Calcium	uncensored	seasonal Kendall	21
18	07308500	Calcium	flow-adjusted	seasonal Kendall	21
19	07308500	Chloride	uncensored	seasonal Kendall	21
20	07308500	Chloride	flow-adjusted	seasonal Kendall	21
21	07316000	Calcium	uncensored	seasonal Kendall	21
22	07316000	Calcium	flow-adjusted	seasonal Kendall	21
23	07316000	Chloride	uncensored	seasonal Kendall	21
24	07316000	Chloride	flow-adjusted	seasonal Kendall	21
25	07331600	Calcium	uncensored	seasonal Kendall	21
26	07331600	Calcium	flow-adjusted	seasonal Kendall	21
27	07331600	Chloride	uncensored	seasonal Kendall	21
28	07331600	Chloride	flow-adjusted	seasonal Kendall	21
29	07336820	Calcium	uncensored	seasonal Kendall	21
30	07336820	Calcium	flow-adjusted	seasonal Kendall	21
31	07336820	Chloride	uncensored	seasonal Kendall	21
32	07336820	Chloride	flow-adjusted	seasonal Kendall	21
33	07342500	Calcium	uncensored	seasonal Kendall	21
34	07342500	Calcium	flow-adjusted	seasonal Kendall	21
35	07342500	Chloride	uncensored	seasonal Kendall	21
36	07342500	Chloride	flow-adjusted	seasonal Kendall	21
37	07343000	Calcium	uncensored	seasonal Kendall	21
38	07343000	Calcium	flow-adjusted	seasonal Kendall	21

39	07343000	Chloride	uncensored	seasonal	Kendall	21
40	07343000	Chloride	flow-adjusted	seasonal	Kendall	21
41	07343200	Calcium	uncensored	seasonal	Kendall	21
42	07343200	Calcium	flow-adjusted	seasonal	Kendall	21
43	07343200	Chloride	uncensored	seasonal	Kendall	21
44	07343200	Chloride	flow-adjusted	seasonal	Kendall	21
45	07343500	Calcium	uncensored	seasonal	Kendall	21
46	07343500	Calcium	flow-adjusted	seasonal	Kendall	21
47	07343500	Chloride	uncensored	seasonal	Kendall	21
48	07343500	Chloride	flow-adjusted	seasonal	Kendall	21
49	07344500	Calcium	uncensored	seasonal	Kendall	21
50	07344500	Calcium	flow-adjusted	seasonal	Kendall	21
51	07344500	Chloride	uncensored	seasonal	Kendall	21
52	07344500	Chloride	flow-adjusted	seasonal	Kendall	21
53	07346070	Calcium	uncensored	seasonal	Kendall	21
54	07346070	Calcium	flow-adjusted	seasonal	Kendall	21
55	07346070	Chloride	uncensored	seasonal	Kendall	21
56	07346070	Chloride	flow-adjusted	seasonal	Kendall	21
	NumSeas	Nobs	RepValue	Trend	Trend.pct	P.value
1	6	123	100.00001	0.00000000	0.000000000	0.8336014152
2	6	123	100.00001	-0.03460094	-0.034600935	0.9760057926
3	6	124	469.89363	12.79590098	2.723148421	0.0336792469
4	6	124	469.89363	12.65228904	2.692585771	0.0129020214
5	6	111	110.00001	2.39588639	2.178078333	0.0003365278
6	6	111	110.00001	1.28664998	1.169681687	0.0130635500
7	6	111	660.00008	17.25881435	2.614971565	0.0007134676
8	6	111	660.00008	11.34893417	1.719535280	0.0029419661
9	6	122	530.00011	1.38744659	0.261782321	0.8355770111
10	6	122	530.00011	3.98002743	0.750948414	0.2728573084
11	6	122	3000.00035	30.77328679	1.025776108	0.8205802441
12	6	122	3000.00035	82.11083142	2.737027397	0.0758367777
13	6	123	480.00003	0.00000000	0.000000000	0.8724372387
14	6	123	480.00003	-0.04248080	-0.008850166	0.9617801309
15	6	123	270.00002	2.26912466	0.840416463	0.0046260357
16	6	123	270.00002	2.25616524	0.835616678	0.0180431604
17	6	119	340.00001	-0.87151697	-0.256328516	0.6151251197
18	6	119	340.00001	-1.02598287	-0.301759663	0.5201364756
19	6	119	1837.99988	8.02723876	0.436737719	0.5162076950
20	6	119	1837.99988	-3.36533913	-0.183097897	0.8067187071
21	12	182	200.00001	0.69571531	0.347857631	0.5388302803
22	12	182	200.00001	2.11190914	1.055954503	0.0956373215
23	12	188	1000.00010	9.27514624	0.927514535	0.3566403389
24	12	188	1000.00010	18.20233758	1.820233583	0.0486334562
25	12	200	88.49858	0.62565134	0.706962055	0.2394109964
26	12	200	88.49858	0.38255860	0.432276569	0.3169723749
27	12	203	320.00001	2.57714511	0.805357813	0.5003621578
28	12	203	320.00001	0.68904602	0.215326871	0.7803473473
29	12	185	65.00001	0.84057835	1.293197225	0.0961654186
30	12	185	65.00001	0.66315946	1.020245132	0.1187556982
31	12	188	166.49328	4.10125215	2.463313924	0.0698596239
32	12	188	166.49328	3.75690997	2.256493459	0.0728764534

33	12	183	40.00000	-0.35416765	-0.885419081	0.2286658287
34	12	183	40.00000	-0.30565923	-0.764148057	0.1911620051
35	12	183	16.00000	-0.31416803	-1.963550197	0.1585604995
36	12	183	16.00000	-0.16349055	-1.021815937	0.2915501893
37	12	170	79.49842	0.20723170	0.260673977	0.6322352886
38	12	170	79.49842	-0.18586967	-0.233802961	0.6531360745
39	12	170	35.00000	-0.16173531	-0.462100829	0.7745313048
40	12	170	35.00000	-0.57620676	-1.646304859	0.0628329962
41	12	184	55.99999	-0.40639101	-0.725698359	0.2162646502
42	12	184	55.99999	-0.57474869	-1.026337109	0.0382853523
43	12	184	21.00000	-0.58466480	-2.784117838	0.0263182856
44	12	184	21.00000	-0.53513019	-2.548238801	0.0054894532
45	12	179	14.00000	-0.07162687	-0.511620472	0.5025991797
46	12	179	14.00000	-0.05716741	-0.408338639	0.3829599321
47	12	179	20.00000	-0.32744925	-1.637246215	0.2295798212
48	12	179	20.00000	-0.32282613	-1.614130573	0.0948637649
49	12	186	16.00000	-0.09071358	-0.566959859	0.2795946300
50	12	186	16.00000	-0.12612523	-0.788282700	0.0304778069
51	12	186	44.49719	-0.49735710	-1.117726880	0.3637532592
52	12	186	44.49719	-0.94321269	-2.119712730	0.0002551555
53	12	183	7.00000	-0.04453901	-0.636271629	0.0876174122
54	12	183	7.00000	-0.05667081	-0.809582982	0.0062053259
55	12	183	30.00000	-0.70519603	-2.350653317	0.0265707411
56	12	183	30.00000	-0.71812751	-2.393758259	0.0005073280

Trend.dir

1	*
2	none
3	up
4	up
5	up
6	up
7	up
8	up
9	none
10	none
11	none
12	none
13	*
14	none
15	up
16	up
17	none
18	none
19	none
20	none
21	none
22	none
23	none
24	up
25	none
26	none

27	none
28	none
29	none
30	none
31	none
32	none
33	none
34	none
35	none
36	none
37	none
38	none
39	none
40	none
41	none
42	down
43	down
44	down
45	none
46	none
47	none
48	none
49	none
50	down
51	none
52	down
53	none
54	down
55	down
56	down

7 Further Remarks

Because trend analysis is not necessarily a straightforward process, but requires user assessments at several points in the process, it is not necessarily a good idea to simply create scripts and run them without any user review and interaction. To overcome recording the steps in a script, the functions in `restrend` record all changes to the projects database in a list called `estrend.cl`. It can be viewed at any time simply by entering `estrend.cl` in the console window. It can be saved with the data to ensure that the trend analysis is reproducible.

```
> # get the history
> estrend.cl
```

```
[[1]]
setProj(project = "majors", data = Majors, STAID = "STAID", DATES = "DATES",
        Snames = c("Calcium", "Chloride"), FLOW = "FLOW", type = "seasonal",
        Start = "1968-10-01", End = "1989-10-01")
```

```
[[2]]
flowAdjust()
```

```
[[3]]
flowAdjust(Stations = "07331600", Snames = c("Calcium", "Chloride"),
          span = 1)
```

```
[[4]]
SKTrends()
```

References

- [1] Lorenz, D.L., in preparation, `restrend`: an R package for ESTimate TRENDS: U.S. Geological Survey Open File Report, ? p.
- [2] Schertz, T.L., Alexander, R.B., and Ohe, D.J., 1991, The computer program ESTimate TREND (ESTREND), a system for the detection of trends in water-quality data: U.S. Geological Survey Water Resources Investigations Report 91-4040, 72 p.