

SPECIAL PUBLICATION SJ2008-SP20

**CLUSTER ANALYSIS AND PRINCIPLE COMPONENT
ANALYSIS OF WATER QUALITY DATA
FROM SILVER SPRINGS AND WELLS
IN THE SPRING SHED**



**Cluster Analysis and
Principle Component Analysis
Of Water Quality Data
From Silver Springs and Wells
In The Spring Shed**

**Report Prepared by Insightful Corp
for St. John's River Watershed Management District**

**Insightful Contacts
Michael O'Connell, moconnell@insightful.com
Bob Treder, bob@insightful.com**

**St. John's River Watershed Management District Contact
David Toth, dtoth@sjrwmd.com**

Executive Summary

Levels of nitrate-N (nitrate) and 12 other water quality parameters were examined from published data from 56 wells in the Silver Springs springshed in an effort to locate specific regions that may be sources of elevated nitrate concentrations in Silver Springs discharge. Clustering, principal components and spatial linear models (kriging) were computed and results plotted to aid in the location of elevated nitrate. A descriptive plot coloring wells by nitrate level and predictions from a kriging spatial model were by far the best locators of regions with elevated nitrate. Three regions were identified as having elevated nitrate levels, hence indicating possible sources of nitrate contamination: one area 6 miles northwest of Silver Springs, and two areas 8 miles and 14 miles south of Silver Springs. The latter area is near the southern boundary of Marion County.

The use of clustering and principal components provide interesting detail on how the 13 water quality parameters are interrelated and specifically which parameters co-vary with nitrate, providing a nitrate-related index to water quality. However, kriging is the best of the three methods used to identify areas contributing to nitrate contamination at Silver Springs.

Table of Contents

Executive Summary	2
Table of Contents	2
1 Introduction	2
2 Objective.....	3
3 Procedures	3
4 Results	4
4.1 Descriptive Statistics	4
4.2 Clustering	7
4.3 Principal Components	9
4.4 Nitrate Spatial Model	13
5 Conclusions	16
6 References	16
7 Appendix A: Well Locations.....	16
8 Appendix B: Understanding A Stem and Leaf Plot	18
9 Appendix C: The S-PLUS Code.....	19

1 Introduction

Silver Springs is the largest spring in the St Johns River Water Management District (District). It consists of 30 different vent groups in the upper 1,200 meters of the Silver River. In 2006, the District sampled the 30 spring vent groups for water quality and nutrients. Most of the different spring vents had elevated (greater than 1.0 mg/L) values

for nitrate-N. This is significant because a value of 1.0 mg/L of nitrate-N is considered by the Florida Springs Task force to be detrimental to the springs' ecology. In 2004, the U. S. Geological Survey (Phelps, 2004) published a report on the chemistry of ground water in the Silver Springs basin with an emphasis on nitrate. Phelps sampled 56 wells located within the Silver Springs spring shed and found nitrate-N to be elevated in many of them. In 2007, the District amended the existing contract with the Florida Department of Environmental Protection (FDEP) for Phase II and III, and to identify the areas in the spring shed that contribute nitrate-N to Silver Springs. The District has chosen cluster analysis and principle component analysis as a means of performing this work. Deliverables from this project will be provided to FDEP as part of the existing agreement between the District and FDEP.

2 Objective

The objective of this study is to identify the discrete areas in the spring shed that contribute nitrate-N to the different vent groups at Silver Springs. To achieve this objective, statistical cluster analysis and principle component analysis will be performed (using SPLUS statistical software) on water quality data collected from 34 samples collected at Silver Springs and 56 wells sampled by Phelps (2004). The Consultant shall prepare a report identifying areas within the spring shed that contribute nitrate-N (nitrate) to the springs. The report shall fully explain the analysis steps used to identify the contributing areas.

3 Procedures

Data received from the staff of the St John's River Water Management District (District) were imported into S-PLUS and examined for irregularities by

1. computing summaries (min, max, mean , median, 25th and 75th percentiles and the number of missing values),
2. creating descriptive plots including histograms and pair-wise scatter plots, and
3. creating maps of well and spring locations colored to indicate elevated values of nitrate.

Because some of the nitrate levels were high (> 5.0 mg/L) all the nitrate values were checked against the data as listed in the Scientific Investigations Report 2004-5144, titled Chemistry of Ground Water in the Silver Springs Basin, Florida, with an Emphasis on Nitrate.

Following the initial examination of the data, clustering (k-means) and principal components analyses were computed in an attempt to identify regions in the Silver Springs spring shed that contribute nitrate and other correlated water quality parameters to the springs. Prior to computing these analyses the data were rescaled to a mean of 0 (zero) and variance of 1 (one) to avoid a potential problem of one or a few of the water quality parameters dominating, due to difference in scale, the clustering and principal components results.

In addition a spatial model was developed to create nitrate surface plots for better describing varying levels of nitrate in the spring shed. A kriging model was used to create the nitrate surface plots.

4 Results

The following sections describe the analyses in detail, sequentially.

1. Descriptive statistics – verifying and understanding the data
2. Clustering
3. Principal components
4. Nitrate spatial model

4.1 Descriptive Statistics

Table 1 and Table 2 show the summary statistics for the water parameters of the wells and vents, respectively. Note the large maximum value of 12.0 for nitrate for wells. Both the mean and median are above acceptable levels of 1.0. Although the maximum for the vents is less than 2.0 (1.9), the mean and median for the vents are also above acceptable levels of 1.0 mg/L.

Table 1: Summary of water parameters for wells.

	Minimum	1 st Quartile	Median	Mean	3 rd Quartile	Maximum	Missing
DO	0.1	1.3	4.1	3.7	5.5	8.3	2
pHfield	6.3	7.0	7.3	7.3	7.5	9.8	0
SpC lab	170.0	324.8	435.0	444.0	523.5	1270.0	0
Temp C	21.8	23.2	23.8	23.8	24.4	25.5	0
Calcium	24.0	48.8	71.5	71.4	88.5	176.0	0
Magnesium	1.1	3.0	6.0	7.4	11.0	25.0	0
Potassium	0.1	0.3	0.6	0.9	1.0	5.6	0
Sodium	2.2	4.3	6.5	11.2	9.2	140.0	0
Alkalinity	29.0	109.0	151.0	154.3	193.8	285.0	0
Chloride	3.0	8.0	9.8	17.7	15.2	230.0	0
Sulfate	0.2	6.0	20.5	47.2	65.0	400.0	0
NOxN	0.0	0.6	1.2	1.7	1.8	12.0	0
Phosphorus Tot	0.0	0.0	0.0	0.1	0.0	2.7	0

Table 2: Summary of water parameters for vents.

	Minimum	1 st Quartile	Median	Mean	3 rd Quartile	Maximum	Missing
DO	0.1	1.9	2.7	2.5	3.5	3.7	0
pHfield	7.0	7.3	7.4	7.4	7.5	7.6	0
SpClab	320.0	357.0	409.0	411.5	466.5	514.0	0
Temp C	22.0	23.4	23.7	23.5	23.7	23.8	0
Calcium	47.0	56.5	66.5	65.8	76.5	85.0	0
Magnesium	6.1	8.1	9.4	9.6	11.0	13.0	0
Potassium	0.5	0.7	0.8	0.8	0.8	0.9	0

Sodium	5.3	6.3	6.7	6.6	7.0	7.2	0
Alkalinity	114.0	137.8	162.5	158.9	177.0	206.0	0
Chloride	9.0	10.7	11.3	11.0	11.5	11.8	0
Sulfate	16.7	33.6	44.0	46.2	59.1	74.4	0
NOxN	0.9	1.3	1.4	1.4	1.5	1.9	0
Phosphorus Tot	0.0	0.0	0.0	0.0	0.0	0.0	0

Wells with nitrate-N levels elevated above 5.0 mg/L are listed in Table 3.

Table 3: Wells with high nitrate levels.

Well Number	NO _x N Level
1	6.3
11	9.9
16	5.1
48	5.9
49	12.0

In addition to the summary tables presented above, it's helpful to look at stem and leaf plots of nitrate-N levels to get a sense of the distribution of those values. A stem and leaf plots displays all the values in the form of a text-based histogram. The values are represented by *stems* to the left of the colon (the decimal point in the plot below) and the leaves (the fractional part of the number). For a detailed description of a stem and leaf plot see Appendix B: Understanding a Stem and Leaf Plot.

N = 56 Median = 1.2
 Quartiles = 0.51, 1.9

Decimal point is at the colon

```

0 : 0000000123444
0 : 66788899
1 : 0111122222444
1 : 5567788
2 : 022244
2 : 7
3 : 1
3 :
4 : 0
High: 5.1 5.9 6.3 9.9 12.0
  
```

Figure 1: Stem and leaf plot for NO_xN levels in the 56 wells.

The first row in Figure 1 corresponds to 8 wells with values of **0.0**, three wells with values of **0.1**, **0.2** and **0.3** and three wells each with values **0.4**. The high values are listed separately at the bottom of the graph.

With the assistance of the stem and leaf plot of Figure 1, four groups of nitrate were created and used to differentially color a map of the wells to get a sense of regions with high nitrate-N levels. The nitrate-N groups are specified with

Table 4: Nitrate-N groups for coloring the wells in Figure 2.

```

NOxN < 0.5
0.5 ≤ NOxN < 1.0
1.0 ≤ NOxN < 2.0
2.0 ≤ NOxN
  
```

The resulting map is displayed in Figure 2. It's clear from the map that most of the wells west and south of the vents have elevated nitrate-N levels. However, there are also two wells north and slightly west of the vents with elevated nitrate-N levels.

Well Colors Determined by NOxN Level

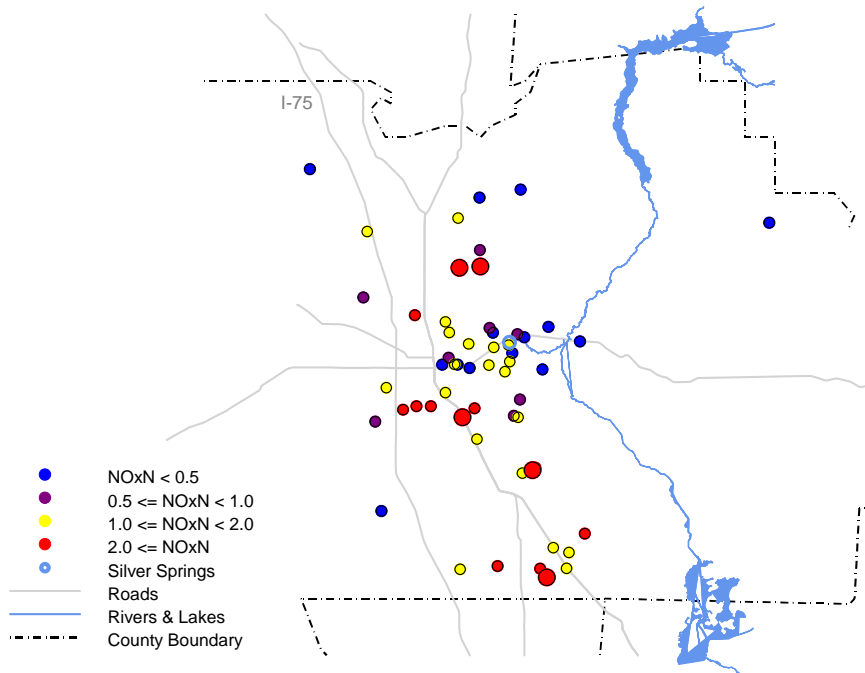


Figure 2: Well locations colored by nitrate-N levels. The five larger red points correspond to nitrate-N levels greater than 5.0 mg/L, those listed in Table 3.

4.2 Clustering

The groups defined by Table 4 provide a suitable starting point for k-means clustering on the water quality parameters displayed in Table 1 and Table 2. By computing the mean level of each parameter for each group we can see clear and consistent differences in some of the parameters. The values for **DO** and **NOxN** increase with group number and those for **Alkalinity** decrease.

Table 5: Mean levels of the water quality parameters for the wells for each of the nitrate groups formed by Table 4.

	Group 1	Group 2	Group 3	Group 4
DO	1.74	2.73	4.2	5.04
pHfield	7.18	7.37	7.32	7.46
SpClab	492.43	343.88	419.2	488.36
TempDegC	23.39	23.54	24.04	23.91
Calcium	82.71	62.88	66.9	71.5
Magnesium	9.81	4.5	8.2	5.54
Potassium	0.96	0.51	0.64	1.55
Sodium	9.36	7.41	7.33	20.8
Alkalinity	168.43	151.5	149.95	148.14
Chloride	14.84	10.44	10.73	34.72
Sulfate	74.09	11.89	49.16	37.64
NOxN	0.14	0.78	1.36	4.46
PhosphorusTotal	0.24	0.05	0.03	0.08

Four clusters were created using *scaled* water quality parameters. The scaled parameters were computed by subtracting the mean and dividing by the standard deviation for each column. Initial group centers used to start the clustering algorithm were computed as the means for all the scaled water quality parameters for each group. The final four clusters had scaled water quality parameter means as listed in Table 6. Note that nitrate increases in the following order **Group 2**, **Group 1**, **Group 3** and **Group 4**.

Table 6: Mean levels of water quality parameters for each of the k-means cluster groups created from well data only.

	Group 1	Group 2	Group 3	Group 4
DO	-0.10	-0.56	0.85	0.84
pHfield	-0.17	-0.48	0.92	-0.97
SpClab	0.74	-0.17	-0.77	4.29
TempDegC	0.28	-0.50	0.38	0.97
Calcium	0.79	0.12	-1.05	1.41
Magnesium	1.11	-0.48	-0.39	-0.44
Potassium	0.32	-0.39	0.28	-0.56
Sodium	-0.04	-0.23	-0.04	6.52
Alkalinity	0.23	0.54	-1.12	1.92
Chloride	-0.06	-0.25	0.01	6.62
Sulfate	1.07	-0.47	-0.37	-0.42
NOxN	-0.20	-0.35	0.59	1.85
PhosphorusTotal	-0.18	0.20	-0.10	-0.11

Similar to Figure 2, well locations were plotted with colors (blue, purple, yellow and red) representing increasing nitrate levels. The pattern is similar to that seen in Figure 2, although not as striking. This is due, no doubt, to some dissipation of the nitrate signal when looking at all the water parameters.

Well Colors Determined by K-Means Clusters

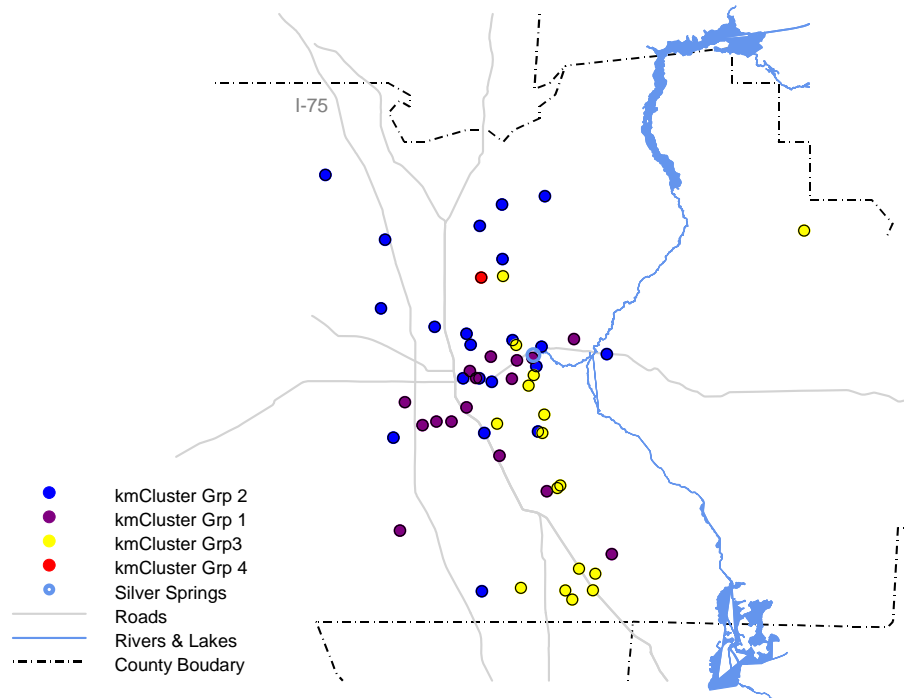


Figure 3: Well locations colored by k-means group membership. Blue, purple, yellow and red show increasing levels of nitrate, respectively.

In addition to k-means clustering, clusters were computed using hierarchical clustering. Hierarchical clustering was less useful in identifying areas of high nitrate because the automated tools for combining observations to condense the grouping did not discriminate nitrate groups very well. Therefore, hierarchical clustering was not further used.

4.3 Principal Components

Principal components for the well data were computed to determine the primary axes of variation among the water parameters. Prior to computing the principal components, the water parameters were rescaled to have mean 0 (zero) and variance 1 (one). Rescaling is critical to avoid a single (or a few) variable(s) from dominating the calculations simply because their scale is much larger than the other parameters. For example, from Table 1, we can see that **SpC lab**, **Calcium**, **Alkalinity**, **Chloride** and **Sulfate** have large scales relative to the other parameters and especially relative to nitrate.

The first six components explain about 90% of the variation with greatly diminished explanatory power after the sixth component. Table 7 lists the coefficients for the first six principal components exceeding 0.3 in absolute value to emphasize those parameters with

the greatest influence on each component. The second component (PC2) includes a large positive coefficient (0.429) for nitrate as well as large positive coefficients for Sodium and Chloride. Consequently large values of PC2 correspond to elevated values of these three parameters.

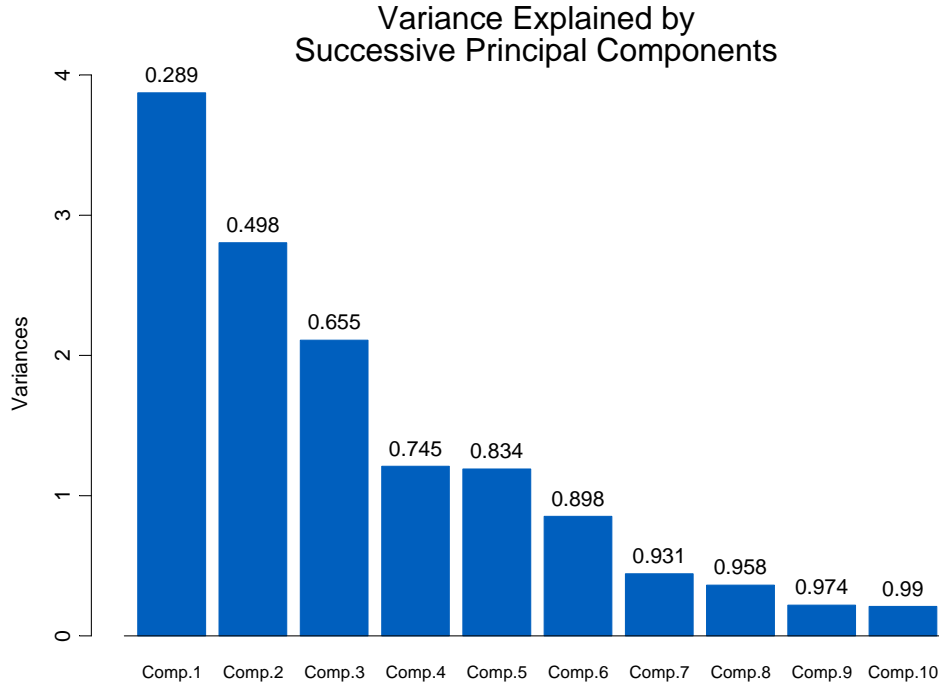


Figure 4: Bar plot of the principal components showing graphically the proportion of variance explained by each component.

Table 7: Coefficients (or loadings) for the first six principal components of the water quality parameters for the wells. Coefficients are shown only if they exceed 0.3 in absolute value to emphasize the parameters having the greatest influence on each component.

	PC 1	PC 2	PC 3	PC 4	PC 5	PC 6
DO					0.453	0.406
pHfield			0.337	-0.436		-0.310
SpClab	-0.455					
Temp C			0.343	0.581		
Calcium	-0.465					
Magnesium	-0.307		0.460			
Potassium			0.317		-0.523	0.351
Sodium		0.517				
Alkalinity	-0.358		-0.419			
Chloride		0.523				
Sulfate	-0.310		0.487			
NOxN		0.429				0.548
Phosphorus Tot				0.531	-0.683	

A plot of the first two principal components with points labeled by the well number is displayed in Figure 5. The four points farthest to the right, 1, 49, 11, and 48 are listed in Table 3 as having high nitrate levels so PC2 identifies those four points. Well 52 is also high on PC2 because it has relatively high Sodium (33) and Chloride (65) levels, two other water quality parameters with high loadings on PC2. The fact that most of the wells cluster in the upper left portion of the plot results from their PC1 and PC2 values being relatively close to zero compared to the extremes.

PC1 vs PC2 With Well Numbers

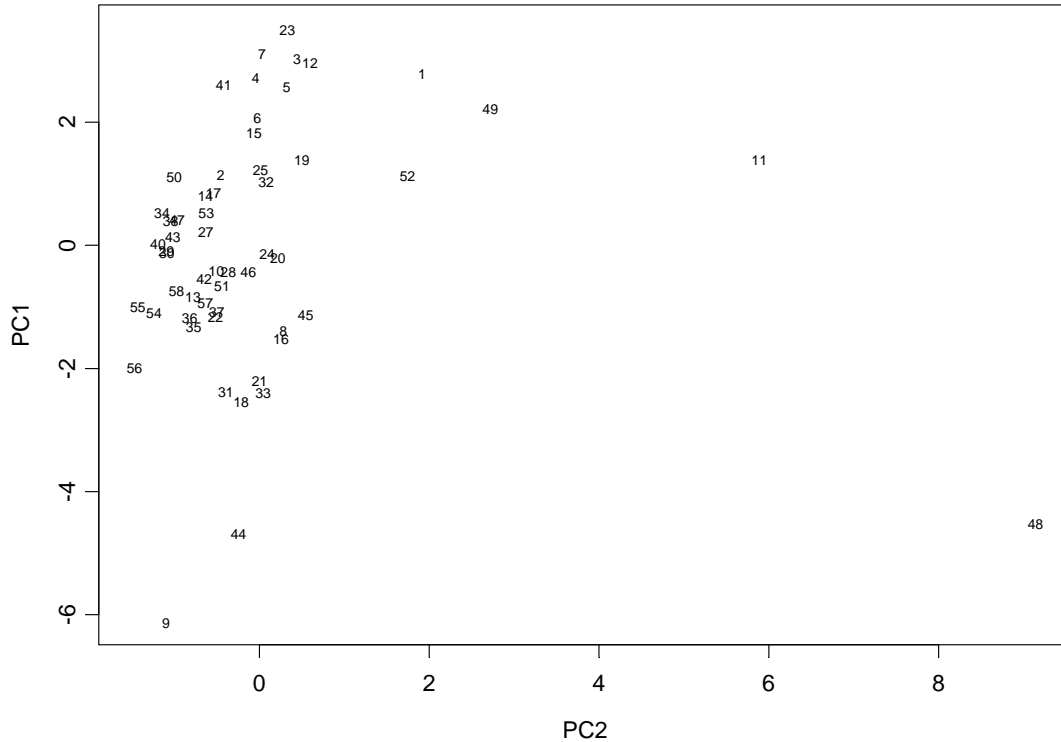


Figure 5: Scatter plot of first two principal components, labeling the points with the well number. Looking at the stem and leaf plot of PC2 with an expanded scale we can create groups and label extreme values of PC2 similar to what was done for Figure 2.

N = 56 Median = -0.408823
 Quartiles = -0.800508, 0.1553385

Decimal point is 1 place to the left of the colon

```

-14 : 73
-13 :
-12 : 4
-11 : 9500
-10 : 9420
-9 : 87
-----break 1
-8 : 2
-7 : 87
-6 : 54332
-5 : 4110
  
```

```

-4 : 5420
-----break 2
-3 : 7
-2 : 41
-1 : 3
-0 : 642
 0 : 013589
-----break 3
 1 :
 2 : 258
 3 : 23
 4 : 4
 5 : 04
 6 : 0
 7 :
 8 :
 9 :
10 :
11 :
12 :
13 :
14 :
15 :
16 :
17 : 5
18 :
19 : 2

```

High: 2.721023 5.888535 9.142356

Figure 6: Stem and leaf plot for the second principal component, PC2, with expanded scale. Note that the decimal point is one place to the left of the colon. Natural breaks occur at -0.8, -0.3 and 0.1 which produces group sizes similar to when grouping on nitrate alone.

Table 8: PC2 groups for coloring the wells in Figure 7.

```

          PC2 < -0.8
-0.8 ≤ PC2 < -0.3
-0.3 ≤ PC2 <  0.1
 0.1 ≤ PC2

```

The resulting map is displayed in Figure 7. Note the similarity between Figure 2 and Figure 7. As seen with the nitrate colored map, higher PC2 values, correlated with higher nitrate levels, occur south and west of the vents.

Well Colors Determined by PC2 Groups

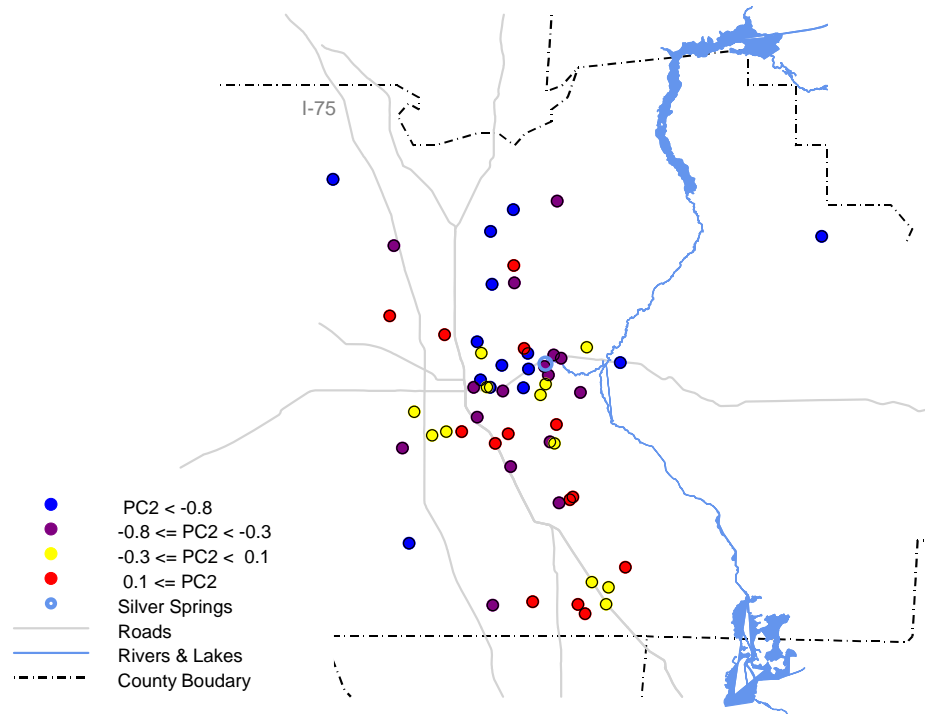


Figure 7: Well locations colored by PC2 groups as specified in Table 8.

4.4 Nitrate Spatial Model

The disadvantage of using clustering or principal components on all the water parameters for locating wells with elevated nitrate levels is that some of the nitrate “signal” or effect is lost when combined with the other parameters. A more direct way of locating elevated levels of nitrate in the spring shed is to model nitrate directly as a spatial model which produces a nitrate *surface* as function of location. This can be done using a kriging model or a spatial linear model. Kriging models based on interpolation have been historically used in mining applications (see, for example, <http://en.wikipedia.org/wiki/Kriging>) by modeling density of mineral and metal deposits based on known density values at various locations in the field under investigation. The mining scenario is completely analogous to our problem of locating regions of high nitrate levels.

A kriging model is a linear interpolation method that allows predictions of unknown values of a *random function* from observations at known locations. See Kaluzny, Vega, Cardoso & Shelly (1998) for more details about kriging models. The random function in our case is a function that describes (or models) the nitrate levels in the St John’s River spring shed. An integral part of a kriging model is the variogram or variance as a function of distance from any given observation. For the kriging model applied to the nitrate levels

we assumed a spherical variogram based on the assumption that a high concentration of nitrate at any given point would diffuse radially and approximately equally in all directions in the ground water. This assumption was confirmed by fitting an empirical variogram based on the data alone and then overlaying the theoretical spherical variogram to verify the fit. The fit was quite good as indicated by Figure 8.

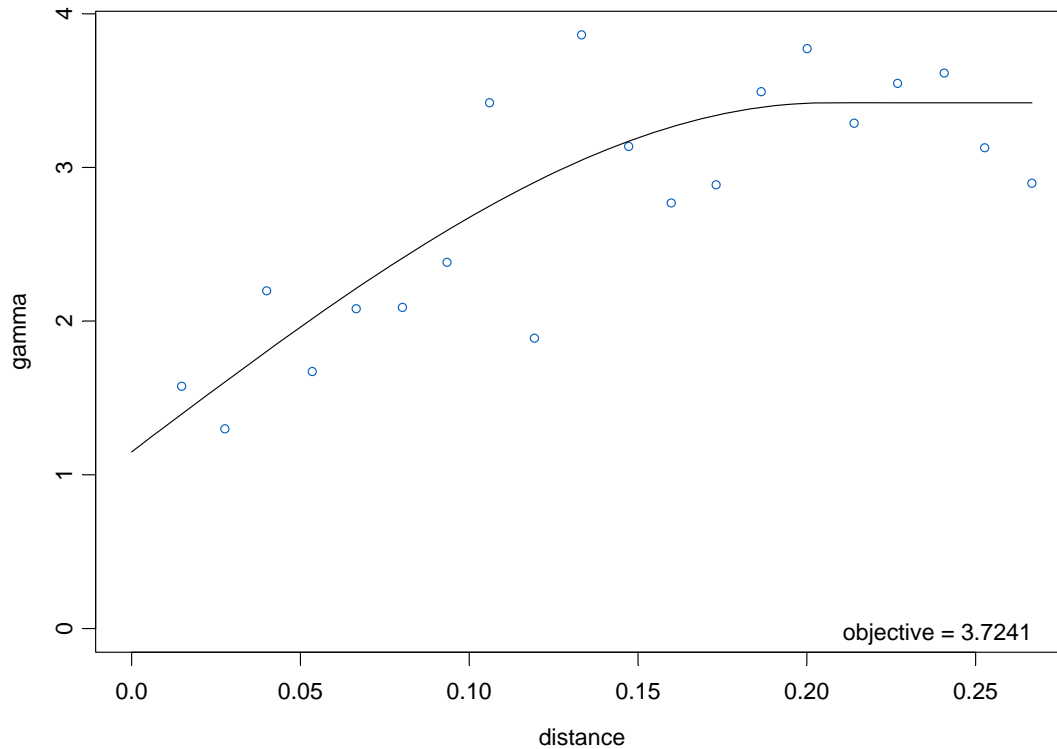


Figure 8: Theoretical variogram overlaying the empirical estimates of variance as a function of distance from any given observation. This graph indicates that a theoretical *spherical* variogram assumption is reasonable for developing a kriging model for nitrate.

Figure 9 displays predictions from a kriging model of nitrate as a function of location. The contours clearly delineate regions of elevated nitrate levels. This model is based solely on the levels of nitrate so there is no lost signal due to combining nitrate with other water parameters. The red triangles on the graph correspond to well locations.

Kriging Predictions

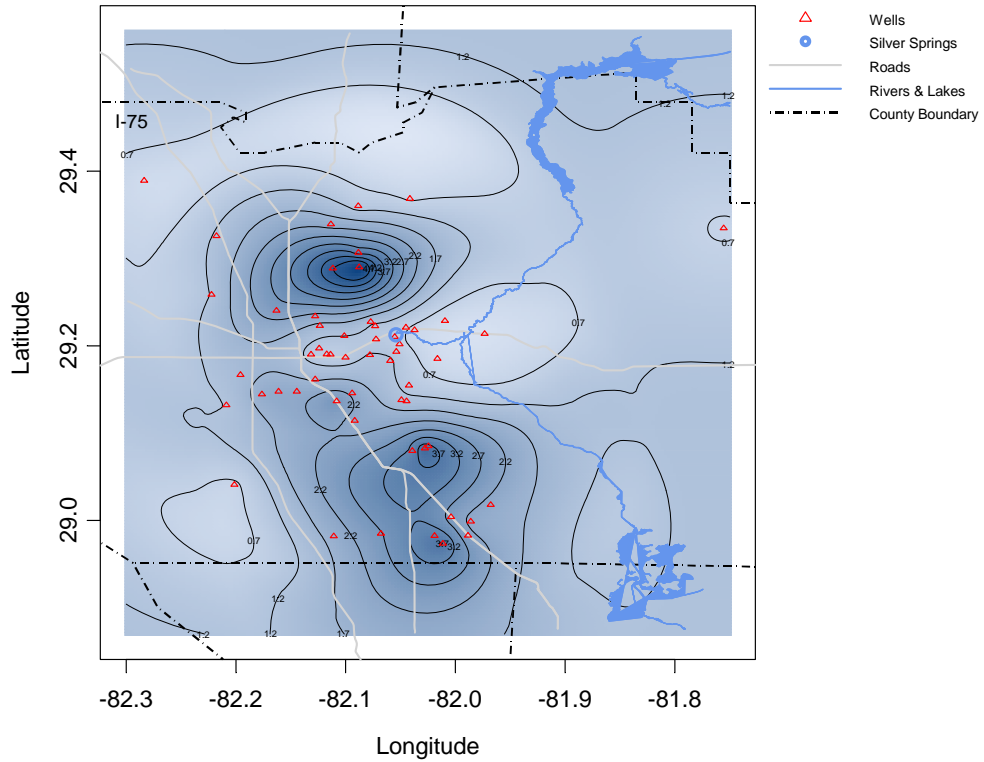


Figure 9: Prediction of nitrate levels based on a spatial (kriging) model. Darker blue corresponds to higher levels of nitrate. Well locations are indicated by the red triangles.

The three wells on or near the northern peak correspond to well numbers 48, 49 and 50. See Table 9 for well locations and nitrate levels. The three wells on or near the southern peak correspond to well numbers 10, 11 and 12. See Table 10 for well locations and nitrate levels. The two wells on or near the peak towards the bottom of the map correspond to well numbers 1 and 3. See Table 11 for well locations and nitrate levels. Table 9, Table 10, and Table 11 combined list five of the six wells with the highest nitrate levels recorded as 4, 5.1, 5.9, 6.3, 9.9 and 12.0.

Table 9: Wells on or near the northern peak of the nitrate surface map.

Well Number	Nitrate value	Longitude	Latitude
48	5.90	-82.113	29.288
49	12.00	-82.088	29.289
50	0.65	-82.088	29.306

Table 10: Wells on or near the southern peak of the nitrate surface map.

Well Number	Nitrate value	Longitude	Latitude
10	1.20	-82.039	29.079
11	9.90	-82.028	29.082
12	2.20	-82.024	29.085

Table 11: Wells on or near the peak towards the bottom of the map.

Well Number	Nitrate value	Longitude	Latitude
1	6.30	-82.011	28.973
3	4.00	-82.019	28.982

5 Conclusions

The primary objective of this study is “to identify the discrete areas in the spring shed that contribute nitrate to the different vent groups at Silver Springs.” The most direct way to accomplish this goal is to locate regions in the Silver Springs spring shed with elevated nitrate levels. This is most easily and directly done by creating the elevated nitrate level plot of Figure 2 and the spatial model prediction surface of Figure 9. Both these displays work directly with the nitrate levels and hence lead to results and conclusions based on nitrate levels alone. In particular, we can see from both figures that elevated levels of nitrate occur to the north and slightly west and to the south of Silver Springs.

Consequently, the most likely sources of nitrate contributing to the spring shed are near wells 48, 49 and 50 to the north and near wells 1, 3, 10, 11 and 12 to the south. These are the same well numbers as used in Phelps (2004).

The use of clustering and principal components provide interesting detail on how the 13 water quality parameters are interrelated and specifically which parameters co-vary with nitrate, providing a nitrate-related index to water quality. However, to specifically answer the question of regions of high nitrate levels contributing to overall spring ill-health they are not as direct or as useful as a spatial model of nitrate alone.

6 References

- Phelps, G.G., 2004, Chemistry of Ground Water in the Silver Springs Basin, Florida, with an Emphasis on Nitrate, U.S. Geological Survey Scientific Investigations Report 2004-5144, 54p.
- Kaluzny, S. P. , Vega, S. C., Cardoso, T. P., Shelly, A. A.1998. *S+Spatial Stats*. Springer, NY, NY. 327 pp.

7 Appendix A: Well Locations

The following table is a list of all the wells with latitude and longitude displayed in both decimal and degree-minute-second (ddmmss) formats. Add NOx data to table 12. Well numbers are the same as in Phelps (2004).

Table 12: Latitude and Longitude for each well in both decimal and degrees, minutes, seconds format.

Well No	Latitude	Longitude	Latitude	Longitude	NOxN
1	28.9731	-82.0114	285823	820041	6.30
2	28.9814	-82.1108	285853	820639	1.80
3	28.9819	-82.0189	285855	820108	4.00
4	28.9822	-81.9886	285856	815919	1.60
5	28.9844	-82.0678	285904	820404	2.40
6	28.9983	-81.9858	285954	815909	1.40

7	29.0033	-82.0039	290012	820014	1.40
8	29.0175	-81.9678	290103	815804	2.00
9	29.0406	-82.2014	290226	821205	0.02
10	29.0792	-82.0392	290445	820221	1.20
11	29.0822	-82.0278	290456	820140	9.90
12	29.0847	-82.0244	290505	820128	2.20
13	29.1139	-82.0917	290650	820530	1.10
14	29.1317	-82.2086	290754	821231	0.93
15	29.1361	-82.0444	290810	820240	1.10
16	29.1361	-82.1083	290810	820630	5.10
17	29.1375	-82.0492	290815	820257	0.94
18	29.1439	-82.1764	290838	821035	2.70
19	29.1453	-82.0942	290843	820539	2.20
20	29.1472	-82.1444	290850	820800	2.20
21	29.1472	-82.1611	290850	820940	2.40
22	29.1611	-82.1278	290900	820700	1.20
23	29.1542	-82.0422	290915	820232	0.85
24	29.1664	-82.1958	290959	821145	1.40
25	29.1825	-82.0594	291057	820334	1.70
26	29.1847	-82.0164	291105	820059	0.02
27	29.1861	-82.1000	291110	820600	0.37
28	29.1892	-82.0778	291121	820440	1.50
29	29.1897	-82.1139	291123	820650	0.02
30	29.1897	-82.1317	291123	820754	0.38
31	29.1900	-82.1172	291124	820702	1.10
32	29.1928	-82.0539	291134	820314	1.80
33	29.1967	-82.1242	291148	820727	0.84
34	29.2014	-82.0508	291205	820303	0.09
35	29.2072	-82.0722	291226	820420	1.10
36	29.2097	-82.0553	291235	820319	1.20
37	29.2108	-82.1011	291239	820604	1.20
38	29.2133	-81.9733	291248	815824	0.02
39	29.2175	-82.0372	291303	820214	0.02
40	29.2206	-82.0450	291314	820242	0.60
41	29.2222	-82.0731	291320	820423	0.42
42	29.2225	-82.1233	291321	820724	1.00
43	29.2269	-82.0772	291337	820438	0.67
44	29.2281	-82.0094	291341	820034	0.20
45	29.2333	-82.1278	291400	820700	1.70
46	29.2400	-82.1631	291424	820947	3.10
47	29.2581	-82.2222	291529	821320	0.77
48	29.2883	-82.1117	291718	820642	5.90
49	29.2897	-82.0878	291723	820516	12.00
50	29.3064	-82.0883	291823	820518	0.65
51	29.3253	-82.2178	291931	821304	1.20
52	29.3342	-81.7556	292003	814520	0.32
53	29.3389	-82.1133	292020	820648	1.50
54	29.3597	-82.0886	292135	820519	0.02
55	29.3681	-82.0414	292205	820229	0.02
56	29.3889	-82.2836	292320	821701	0.05

8 Appendix B: Understanding A Stem and Leaf Plot

In a stem-and-leaf plot each data value is split into a **stem** and a **leaf**. The leaf is usually the last digit of the number and the other digits to the left of the leaf form the stem. The number 123 would be split as:

stem 12
leaf 3

Constructing a stem-and-leaf plot:

The data: Math test scores out of 50 points: 35, 36, 38, 40, 42, 42, 44, 45, 45, 47, 48, 49, 50, 50, 50.

<p>Writing the data in numerical order may help to organize the data, but is NOT a required step. Ordering can be done later.</p>	<p>35, 36, 38, 40, 42, 42, 44, 45, 45, 47, 48, 49, 50, 50, 50</p>										
<p>Separate each number into a stem and a leaf. Since these are two digit numbers, the tens digit is the stem and the units digit is the leaf.</p>	<p>The number 38 would be represented as</p> <table border="1" data-bbox="883 1136 1214 1213"> <thead> <tr> <th>Stem</th> <th>Leaf</th> </tr> </thead> <tbody> <tr> <td>3</td> <td>8</td> </tr> </tbody> </table>	Stem	Leaf	3	8						
Stem	Leaf										
3	8										
<p>Group the numbers with the same stems. List the stems in numerical order. (If your leaf values are not in increasing order, order them now.) Title the graph.</p>	<table border="1" data-bbox="854 1262 1243 1493"> <thead> <tr> <th colspan="2">Math Test Scores (out of 50 pts)</th> </tr> <tr> <th>Stem</th> <th>Leaf</th> </tr> </thead> <tbody> <tr> <td>3</td> <td>5 6 8</td> </tr> <tr> <td>4</td> <td>0 2 2 4 5 5 7 8 9</td> </tr> <tr> <td>5</td> <td>0 0 0</td> </tr> </tbody> </table>	Math Test Scores (out of 50 pts)		Stem	Leaf	3	5 6 8	4	0 2 2 4 5 5 7 8 9	5	0 0 0
Math Test Scores (out of 50 pts)											
Stem	Leaf										
3	5 6 8										
4	0 2 2 4 5 5 7 8 9										
5	0 0 0										
<p>Prepare an appropriate legend (key) for the graph.</p>	<p>Legend: 3 6 means 36</p>										

A stem-and-leaf plot shows the shape and distribution of data. It can be clearly seen in the diagram above that the data clusters around the row with a stem of 4.

9 Appendix C: The S-PLUS Code

Following is the S-PLUS code used to create the tables and figures used in this report.

```
### Load the maps library for create map graphics of Marion county
library(maps)

### read the well data and modify the column names
Phelps = importData("../FilesFromClient/Phelps&SSG-I.xls", stringsAsFactors = F)
names(Phelps) = c("WellNo", "USGSsiteID", "Latitude", "Longitude", "WellDepthFeet",
  "Date", "DO", "pHfield", "SpClab",
  "TempDegC", "Calcium", "Magnesium", "Potassium", "Sodium",
  "Alkalinity", "Chloride", "Sulfate", "NOxN",
  "PhosphorusTotal")

### Function for converting deg/min/sec lat and long data to decimal equivalents
### Verified this function using the on-line conversion port
### http://www.fcc.gov/mb/audio/bickel/DDDMSS-decimal.html
geoDegMinSec2Dec = function(x){
  # compute degrees
  Sgn = sign(x)
  x = abs(x)
  Deg = trunc(x)
  MinSec = (x - Deg) * 100
  Min = trunc(MinSec)
  Sec = (MinSec - Min) * 100
  Sgn * (Deg + Min/60 + Sec/3600)
}

### Create vents variable indicating which observations are vents
vents = substring(Phelps$USGSsiteID, 1, 3) == "SSG"

#####
### Table 1 ###
#####
### Create summary for Report - Table 1
ssWells = summary(Phelps[!vents, ])
print.default(ssWells)
H2OParamsWells = t(ssWells[, 7:19])

lapply(apply(H2OParamsWells, 1, unpaste, sep = ":"), function(x)x[[1]])[[1]]
valuesWells = t(sapply(apply(H2OParamsWells, 1, unpaste, sep = ":"), function(x)x[[2]]))
dimnames(valuesWells)[[2]] = lapply(apply(H2OParamsWells, 1, unpaste, sep = ":"),
  function(x)x[[1]])[[1]]
valuesWells
valWells = apply(valuesWells, 2, as.numeric)
dimnames(valWells)[[1]] = dimnames(valuesWells)[[1]]
valWells = format(round(valWells, 1))
exportData(valWells, "summariesWells.xls")
print(values, quote = F)

#####
### Table 2 ###
#####
### Create summary for Report - Table 2
ssVents = summary(Phelps[vents, ])
print.default(ssVents)
H2OParamsVents = t(ssVents[, 7:19])

lapply(apply(H2OParamsVents, 1, unpaste, sep = ":"), function(x)x[[1]])[[1]]
valuesVents = t(sapply(apply(H2OParamsVents, 1, unpaste, sep = ":"), function(x)x[[2]]))
dimnames(valuesVents)[[2]] = lapply(apply(H2OParamsVents, 1, unpaste, sep = ":"),
  function(x)x[[1]])[[1]]
valuesVents
valVents = apply(valuesVents, 2, as.numeric)
dimnames(valVents)[[1]] = dimnames(valuesVents)[[1]]
valVents = format(round(valVents, 1))
exportData(valVents, "summariesVents.xls")
```

```

### Rescale lat and long
Phelps$Latitude = Phelps$Latitude/10000
Phelps$Longitude = -Phelps$Longitude/10000

### Convert from Deg/Min/Sec to decimal
Phelps$Latitude = geoDegMinSec2Dec(Phelps$Latitude)
Phelps$Longitude = geoDegMinSec2Dec(Phelps$Longitude)

summary(Phelps)

### Number of vents
sum(substring(Phelps$USGSSsiteID, 1, 3) == "SSG")
### Number of wells
sum(substring(Phelps$USGSSsiteID, 1, 3) != "SSG")

#####
### Table 3 ###
#####
### View wells with elevated NOxN levels - Table 3
names(Phelps)
Phelps[Phelps$NOxN >= 5, c("WellNo", "NOxN")]
Phelps[Phelps$NOxN >= 2, c("WellNo", "NOxN")]
Phelps[Phelps$NOxN >= 5, c(1,7:19)]

#####
### Figure 1 ###
#####
### Create stem and leaf plots of NOx-N values separate for vents and wells - Figure 1
stem(Phelps$NOxN[vents], scale = 0)
stem(Phelps$NOxN[!vents], scale = 0)

stem(Phelps$NOxN[vents], scale = 1)
stem(Phelps$NOxN[!vents], scale = 1)

### Look are sorted values of NOx-N values for vents and wells
sort(unique(Phelps$NOxN[vents]))
sort(unique(Phelps$NOxN[!vents]))

### Create state map with single color for wells and vents
map("county", regions = "florida")
points(Phelps$Longitude, Phelps$Latitude, cex = .5)
title("Florida With Wells and Vents Overlaid")
par("usr")

### Create state map with separate colors for wells and vents
map("county", regions = "florida")
points(Phelps$Longitude[!vents], Phelps$Latitude[!vents], col = "blue", cex = .5)
points(Phelps$Longitude[vents], Phelps$Latitude[vents], col = "red", cex = .5)
key(corner = c(0,0), points = list(pch = c(1, 1), col = c("blue", "red")),
    text = list(c("well", "vent")))

summary(Phelps[vents, c("Longitude", "Latitude")])
summary(Phelps[!vents, c("Longitude", "Latitude")])

### Zoom in
map("county", regions = "florida", ylim = c(28.96, 29.4), xlim = c(-82.3, -81.75))
points(Phelps$Longitude[!vents], Phelps$Latitude[!vents], col = "blue")
points(Phelps$Longitude[vents], Phelps$Latitude[vents], col = "red")
key(corner = c(0,0), points = list(pch = c(1, 1), col = c("blue", "red")),
    text = list(c("well", "vent")))

### Jittering the vents to better see multiple vents
map("county", regions = "florida", ylim = c(28.96, 29.4), xlim = c(-82.3, -81.75))
vents = substring(Phelps$USGSSsiteID, 1, 3) == "SSG"
points(jitter(Phelps$Longitude[vents], amount = .05), jitter(Phelps$Latitude[vents],
    amount = .05), col = "red")
points(Phelps$Longitude[!vents], Phelps$Latitude[!vents], col = "blue")
key(corner = c(0,0), points = list(pch = c(1, 1), col = c("blue", "red")),
    text = list(c("well", "vent")))

```

```

### Fill in plotting symbols with color depending on the values of NOx-N
map("county", regions = "florida", ylim = c(28.96, 29.4), xlim = c(-82.3, -81.75))
points(Phelps$Longitude[!vents], Phelps$Latitude[!vents], col = "blue")
points(Phelps$Longitude[!vents & Phelps$NOxN > 1.8], Phelps$Latitude[!vents & Phelps$NOxN
  > 1.8], col = "blue", pch = 16,
  cex = sqrt(Phelps$NOxN[!vents & Phelps$NOxN > 1.8]/12))

points(Phelps$Longitude[vents], Phelps$Latitude[vents], col = "red")
points(Phelps$Longitude[vents & Phelps$NOxN > 1.8], Phelps$Latitude[vents & Phelps$NOxN >
  1.8], col = "red", pch = 16,
  cex = sqrt(Phelps$NOxN[vents & Phelps$NOxN > 1.8]/12))

key(corner = c(0,0), points = list(pch = c(1, 1), col = c("blue", "red")),
  text = list(c("well", "vent")))

### Another view of the NOxN data
### Differentially color points using RGBA specs
java.graph()
map("county", regions = "florida", ylim = c(28.96, 29.4), xlim = c(-82.3, -81.75))
vents = substring(Phelps$USGSSsiteID, 1, 3) == "SSG"
points(Phelps$Longitude[!vents], Phelps$Latitude[!vents], pch = 16, cex = 1,
  col = hsv(h=0.6666667, s = 1, v = 1, alpha = pmin(1, Phelps$NOxN[!vents]/6)))
points(Phelps$Longitude[vents], Phelps$Latitude[vents], pch = 16, cex = 1,
  col = hsv(h=0, s = 1, v = 1, alpha = pmin(1, Phelps$NOxN[vents]/6)))
key(corner = c(0,0), points = list(pch = c(16, 16), col = c("blue", "red")),
  text = list(c("well", "vent")))

#####
### Map With Wells Colored by NOxN Groups ###
#####

### Create grouping on NOxN values - Table 4
stem(Phelps$NOxN)
sort(Phelps$NOxN)
# create NOxN Groups
# grp 1: NOxN < 0.5
# grp 2: 0.5 <= NOxN < 1.0
# grp 3: 1.0 <= NOxN < 2.0
# grp 4: 2.0 <= NOxN

Phelps$NOxNgrp <- rep(1, numRows(Phelps))
Phelps$NOxNgrp[Phelps$NOxN >= 0.5] <- 2
Phelps$NOxNgrp[Phelps$NOxN >= 1.0] <- 3
Phelps$NOxNgrp[Phelps$NOxN >= 2.0] <- 4

### Explore summary statistics for the groups
summary(Phelps[, c("NOxN", "NOxNgrp")])
summary(Phelps$NOxNgrp)
table(Phelps$NOxNgrp)
table(Phelps$NOxNgrp[!vents])

tapply(Phelps$NOxN, Phelps$NOxNgrp, mean)
tapply(Phelps$NOxN, Phelps$NOxNgrp, median)
tapply(Phelps$NOxN, Phelps$NOxNgrp, range)

#####
### Figure 2 ###
#####
### Create wells and vents map - Figure 2
map("county", regions = "florida", ylim = c(28.9, 29.54), xlim = c(-82.25, -81.943), lty
  = 3, lwd = 2)
## add state route
state.routeLatLongOK = state.routeLatLong[104:323, ]
lines( x = state.routeLatLongOK$Longitude, y = state.routeLatLongOK$Latitude,
  col="lightgray", lwd = 2)

## add inter-state route

```

```

lines( x = is.routeLatLong$Longitude, y = is.routeLatLong$Latitude, col="lightgray",
      lwd = 2)
text("I-75", x = -82.28, y = 29.46, adj = 1, col = "grey", cex = .8)
## add us route
us.routeLatlongOK = us.routeLatLong[1:2116, ]
lines( x = us.routeLatLongOK$Longitude, y = us.routeLatLongOK$Latitude, col="lightgray",
      lwd = 2)

## add lakes and rivers
polygon(marionhydroLatLong2$Longitude, marionhydroLatLong2$Latitude,
       border="cornflowerblue", col="cornflowerblue")

## add colored wells
points(Phelps$Longitude[!vents & Phelps$NOxGrp == 1], Phelps$Latitude[!vents &
       Phelps$NOxGrp == 1], col = "blue", pch = 16)
points(Phelps$Longitude[!vents & Phelps$NOxGrp == 2], Phelps$Latitude[!vents &
       Phelps$NOxGrp == 2], col = "purple", pch = 16)
points(Phelps$Longitude[!vents & Phelps$NOxGrp == 3], Phelps$Latitude[!vents &
       Phelps$NOxGrp == 3], col = "yellow", pch = 16)
points(Phelps$Longitude[!vents & Phelps$NOxGrp == 4], Phelps$Latitude[!vents &
       Phelps$NOxGrp == 4], col = "red", pch = 16)
points(Phelps$Longitude[!vents], Phelps$Latitude[!vents])
points(Phelps$Longitude[!vents & Phelps$NOxGrp == 4 & Phelps$NOxN > 5],
       Phelps$Latitude[!vents & Phelps$NOxGrp == 4 & Phelps$NOxN > 5],
       col = "red", pch = 16, cex = 1.5)
points(Phelps$Longitude[!vents & Phelps$NOxN > 5], Phelps$Latitude[!vents & Phelps$NOxN >
       5], cex = 1.5)

### Add Silver Springs
points(silverPoint$Longitude, silverPoint$Latitude, col="cornflowerblue", cex = 1.3, lwd
      = 3)

key(corner = c(0,0), lines = list(pch = c(16, 16, 16, 16, 1, 1, 1, 1),
                                col = c("blue", "purple", "yellow", "red",
                                        "cornflowerblue", "lightgray",
                                        "cornflowerblue", "black"),
                                lwd = c(1, 1, 1, 1, 4, 2, 2, 2),
                                lty = c(rep(1, 7), 3),
                                type = c("p", "p", "p", "p", "p", "l", "l", "l")),
    text = list(c("NOxN < 0.5", "0.5 <= NOxN < 1.0",
                  "1.0 <= NOxN < 2.0", "2.0 <= NOxN",
                  "Silver Springs", "Roads", "Rivers & Lakes",
                  "County Boundary"),
               cex = .75))
title("Well Colors Determined by NOxN Level")

#####
### K-means clustering ###
#####
summary(Phelps)

### Remove missing values from the water parameters
Phelps.noNA = na.exclude(Phelps[, 7:20])
indx.nas = attr(Phelps.noNA, "na.action")

dim(Phelps.noNA) #88 x 14
summary(Phelps.noNA)

### Select the wells
Phelps.cl = scale(Phelps.noNA[!vents, 1:13])
dim(Phelps.cl) #56 x 13
dimnames(Phelps.cl)
class(Phelps.cl)

### Create centers for starting the clustering based on the NOxNgrp groups for wells
grpCenters = matrix(0, ncol = 19 - 7 + 1, nrow = 4)
dimnames(grpCenters) = list(1:4, dimnames(Phelps.cl)[[2]])
for(i in 1:4) grpCenters[i, ] = colMeans(Phelps.cl[Phelps.noNA[!vents,]$NOxNgrp == i, ])

```



```

#####
### Table 5 ###
#####
exportData(t(round(grpCenters, 2)), "grpCenters.xls", rowNames = T) # - Table 5

### Compute k-means clusters
km.Phelps = kmeans(Phelps.cl, centers = grpCenters)

#####
### Table 6 ###
#####
t(round(km.Phelps$centers, 2)) # - Table 6
exportData(t(round(km.Phelps$centers, 2)), "kmGrpCenters.xls", rowNames = T)

summary(km.Phelps)
names(km.Phelps)
km.Phelps$cluster
length(km.Phelps$cluster)
dim(Phelps.cl)
Phelps.cl = cbind(Phelps.cl, kmGrp = km.Phelps$cluster)

#####
### Figure 3 ###
#####
map("county", regions = "florida", ylim = c(28.9, 29.54), xlim = c(-82.25, -81.943), lty
    = 3, lwd = 2)
## add state route
state.routeLatLongOK = state.routeLatLong[104:323, ]
lines( x = state.routeLatLongOK$Longitude, y = state.routeLatLongOK$Latitude,
      col="lightgray", lwd = 2)

## add inter-state route
lines( x = is.routeLatLong$Longitude, y = is.routeLatLong$Latitude, col="lightgray",
      lwd = 2)
text("I-75", x = -82.28, y = 29.46, adj = 1, col = "grey", cex = .8)

## add us route
us.routeLatLongOK = us.routeLatLong[1:2116, ]
lines( x = us.routeLatLongOK$Longitude, y = us.routeLatLongOK$Latitude, col="lightgray",
      lwd = 2)

polygon(marionhydroLatLong2$Longitude, marionhydroLatLong2$Latitude,
      border="cornflowerblue", col="cornflowerblue")

## add wells
points(Phelps[-indx.nas, "Longitude"][!vents[-indx.nas]][Phelps.cl[, "kmGrp"] == 2],
      Phelps[-indx.nas, "Latitude"][!vents[-indx.nas]][Phelps.cl[, "kmGrp"] == 2], col =
      "blue", pch = 16)
points(Phelps[-indx.nas, "Longitude"][!vents[-indx.nas]][Phelps.cl[, "kmGrp"] == 1],
      Phelps[-indx.nas, "Latitude"][!vents[-indx.nas]][Phelps.cl[, "kmGrp"] == 1], col =
      "purple", pch = 16)
points(Phelps[-indx.nas, "Longitude"][!vents[-indx.nas]][Phelps.cl[, "kmGrp"] == 3],
      Phelps[-indx.nas, "Latitude"][!vents[-indx.nas]][Phelps.cl[, "kmGrp"] == 3], col =
      "yellow", pch = 16)
points(Phelps[-indx.nas, "Longitude"][!vents[-indx.nas]][Phelps.cl[, "kmGrp"] == 4],
      Phelps[-indx.nas, "Latitude"][!vents[-indx.nas]][Phelps.cl[, "kmGrp"] == 4], col =
      "red", pch = 16)
points(Phelps[-indx.nas, "Longitude"][!vents[-indx.nas]], Phelps[-indx.nas,
      "Latitude"][!vents[-indx.nas]])

key(corner = c(0,0), lines = list(pch = c(16, 16, 16, 16, 1, 1, 1, 1),
      col = c("blue", "purple", "yellow", "red",
      "cornflowerblue", "lightgray", "cornflowerblue", "black"),
      lwd = c(1, 1, 1, 1, 4, 2, 2, 2),
      lty = c(rep(1, 7), 3),
      type = c("p", "p", "p", "p", "p", "l", "l", "l")),
      text = list(c("NOxN < 0.5", "0.5 <= NOxN < 1.0", "1.0 <= NOxN <
      2.0", "2.0 <= NOxN", "Silver Springs", "Roads", "Rivers & Lakes", "County Boundary"),
      cex = .75))
title("Well Colors Determined by K-Means Clusters")

```

```

### Add Silver Springs
points(silverPoint$Longitude, silverPoint$Latitude, col="cornflowerblue", cex = 1.3, lwd
      = 3)

#####
### Heirarchical Clustering ### # not used
#####

class(Phelps.cl)
hc.Phelps = hclust(dist(Phelps.cl))
plOut = plclust(hc.Phelps, cex = .5)
title("hclust on Phelps.cl")
names(hc.Phelps)

hc.Phelps$height
hist(hc.Phelps$height, n = 15)
cbind(1:88, hc.Phelps$order, c(0, hc.Phelps$height))

# create groups
table(cutree(hc.Phelps, h = 5))
ck = cbind(id = seq(numRows(Phelps.cl)), NOxN = Phelps.cl[, "NOxN"], NOxN2 =
  Phelps.noNA$NOxN, treeGroups = cutree(hc.Phelps, h = 5))
tapply(ck[, "NOxN2"], ck[, "treeGroups"], mean)

plclust(hc.Phelps, hmin = 5, cex = .5)
Phelps = cbind(Phelps, hcGrp = factor(cutree(hc.PhelpsPS, h = 5.3)))
Phelps[Phelps[, "hcGrp"] >= 5, "hcGrp"] = 5
table(Phelps[, "hcGrp"])
sapply(Phelps, class)
tapply(Phelps$NOxN, Phelps$hcGrp, mean)
tapply(Phelps$NOxN, Phelps$hcGrp, median)
tapply(Phelps$NOxN, Phelps$hcGrp, range)
for(i in levels(Phelps$hcGrp)) print(stem(Phelps[Phelps$hcGrp == i, "NOxN"]))

cbind(c(48, 23, 1, 8, 11, 49, 9, 39, 44, 30), Phelps[c(48, 23, 1, 8, 11, 49, 9, 39, 44,
  30), "NOxN"])

summary(PhelpsPS)
dim(Phelps)
cutree(hc.PhelpsPS, h = 4)

#####
### Principal Components ###
#####
pairs(Phelps[!vents, 7:19])
stem(Phelps[!vents, "NOxN"])
stem(Phelps[!vents, "NOxN"], scale = 1)
apply(Phelps.cl, 2, summary)

### Compute principal components
pc.Phelps.cl = princomp(Phelps.cl)
attributes(pc.Phelps.cl)
summary(pc.Phelps.cl)

#####
### Figure 4 ###
#####
screplot(pc.Phelps.cl, cex = .8, main = "Variance Explained by\nSuccessive Principal
  Components")
loadings(pc.Phelps.cl)

#####
### Table 7 ###
#####
print(loadings(pc.Phelps.cl), cutoff = .3)

pc.Phelps.cl$scores

```

```

dimnames(pc.Phelps.cl$scores)
dim(pc.Phelps.cl$scores)
summary(pc.Phelps.cl$scores[, 2])

#####
### Figure 5 ###
#####
PC1 = pc.Phelps.cl$scores[, 1]
PC2 = pc.Phelps.cl$scores[, 2]
plot(PC2, PC1)

plot(PC2, PC1, type = "n")
text(PC2, PC1, dimnames(pc.Phelps.cl$scores)[[1]], cex = .6, col = "black")
title("PC1 vs PC2 With Well Numbers")

pairs(pc.PhelpsPS$scores[, 1:6])

#####
### Figure 6 ###
#####
## histogram for PC2
hist(PC2, n = 20)
stem(PC2, scale = 1)
stem(PC2)

#####
### Table 8 ###
#####
# PC2 < -.80          # blue
# -.80 <= PC2 < -.30 # green
# -.30 <= PC2 < 0.10 # yellow
# 0.10 <= PC2        # red

# or
# PC2 < 2.2
# 2.2 <= PC2 < 3.0
# 3.0 <= PC2 < 3.5
# 3.5 <= PC2

dim(Phelps.cl)
dimnames(Phelps.cl)

#####
### Figure 7 ###
#####
#Phelps.cl = Phelps.cl[, 1:14]
Phelps.cl <- cbind(Phelps.cl, PC2grp = rep(1, numRows(Phelps.cl)))
Phelps.cl[, "PC2grp"][PC2 >= -0.80] <- 2
Phelps.cl[, "PC2grp"][PC2 >= -0.30] <- 3
Phelps.cl[, "PC2grp"][PC2 >= 0.10] <- 4
table(Phelps.cl[, "PC2grp"])

PCgrpCenters = matrix(0, ncol = 19 - 7 + 1, nrow = 4)
dimnames(PCgrpCenters) = list(1:4, dimnames(Phelps.cl)[[2]][1:13])
for(i in 1:4) PCgrpCenters[i, ] = colMeans(Phelps.cl[, 1:13][Phelps.cl[, "PC2grp"] == i,
])

map("county", regions = "florida", ylim = c(28.9, 29.54), xlim = c(-82.25, -81.943), lty
= 3, lwd = 2)
## add state route
state.routeLatLongOK = state.routeLatLong[104:323, ]
lines( x = state.routeLatLongOK$Longitude, y = state.routeLatLongOK$Latitude,
col="lightgray", lwd = 2)

## add inter-state route
lines( x = is.routeLatLong$Longitude, y = is.routeLatLong$Latitude, col="lightgray",
lwd = 2)
text("I-75", x = -82.28, y = 29.46, adj = 1, col = "grey", cex = .8)

```

```

## add us route
us.routeLatLongOK = us.routeLatLong[1:2116, ]
lines( x = us.routeLatLongOK$Longitude, y = us.routeLatLongOK$Latitude, col="lightgray",
      lwd = 2)

## add lakes and rivers
polygon(marionhydroLatLong2$Longitude, marionhydroLatLong2$Latitude,
      border="cornflowerblue", col="cornflowerblue")

## add wells
points(Phelps[!vents, "Longitude"][Phelps.cl[, "PC2grp"] == 1], Phelps[!vents,
  "Latitude"][Phelps.cl[, "PC2grp"] == 1], col = "blue", pch = 16)
points(Phelps[!vents, "Longitude"][Phelps.cl[, "PC2grp"] == 2], Phelps[!vents,
  "Latitude"][Phelps.cl[, "PC2grp"] == 2], col = "purple", pch = 16)
points(Phelps[!vents, "Longitude"][Phelps.cl[, "PC2grp"] == 3], Phelps[!vents,
  "Latitude"][Phelps.cl[, "PC2grp"] == 3], col = "yellow", pch = 16)
points(Phelps[!vents, "Longitude"][Phelps.cl[, "PC2grp"] == 4], Phelps[!vents,
  "Latitude"][Phelps.cl[, "PC2grp"] == 4], col = "red", pch = 16)
points(Phelps[!vents, "Longitude"], Phelps[!vents, "Latitude"])

key(corner = c(0,0), lines = list(pch = c(16, 16, 16, 16, 1, 1, 1, 1),
  col = c("blue", "purple", "yellow", "red",
  "cornflowerblue", "lightgray", "cornflowerblue", "black"),
  lwd = c(1, 1, 1, 1, 4, 2, 2, 2),
  lty = c(rep(1, 7), 3),
  type = c("p", "p", "p", "p", "p", "l", "l", "l")),
  text = list(c(" PC2 < -0.8", "-0.8 <= PC2 < -0.3", "-0.3 <= PC2 <
  0.1", " 0.1 <= PC2", "Silver Springs", "Roads", "Rivers & Lakes", "County Boudary"),
  cex = .75))
title("Well Colors Determined by PC2 Groups")

### Add Silver Springs
points(silverPoint$Longitude, silverPoint$Latitude, col="cornflowerblue", cex = 1.3, lwd
  = 3)

#####
### Spatial Model ###
#####
module(spatial)
NOxN.var1 = variogram(log(NOxN) ~ loc(Longitude, Latitude), data = Phelps,
  subset = !vents, na.action = na.exclude)

## use parameter estimates from the following fit for the krige fit.
variogram.fit(NOxN.var1)

## fit kriging model
Phelps.krig2 = krige(NOxN ~ loc(Longitude, Latitude, angle = 0, ratio = 1),
  data = Phelps, subset = !vents,
  na.action = na.exclude, covfun = spher.cov,
  range = 0.205796078261651,
  sill = 2.27028118686959,
  nugget = 1.14960423109506)

#Compute kriging predictions
Long.seq = seq(-82.3, -81.75, length = 200)
Lat.seq = seq(28.87, 29.56, length = 200)
pgrid = expand.grid(Longitude = Long.seq, Latitude = Lat.seq)
pred = predict(Phelps.krig2, newdata = pgrid)

## Plot the surface
opar = par(pty = "s")
image(x = Long.seq, y = Lat.seq, z = mfit)
par(xaxs = "d")
contour(x = Long.seq, y = Lat.seq, z = mfit, add = T, nlevels = 10, labex = .4)

## Add country boundary
map("county", regions = "florida", ylim = c(28.87, 29.56), xlim = c(-82.3, -81.75),
  add = T, lwd = 2, lty = 3)

```

```

## add state route
state.routeLatLongOK = state.routeLatLong[104:323, ]
lines( x = state.routeLatLongOK$Longitude, y = state.routeLatLongOK$Latitude,
       col="lightgray", lwd = 2)

## add inter-state route
lines( x = is.routeLatLong$Longitude, y = is.routeLatLong$Latitude,   col="lightgray",
       lwd = 2)

## add us route
us.routeLatlongOK = us.routeLatLong[1:2116, ]
lines( x = us.routeLatLongOK$Longitude, y = us.routeLatLongOK$Latitude, col="lightgray",
       lwd = 2)

## Add lakes and rivers
polygon(marionhydroLatLong2$Longitude, marionhydroLatLong2$Latitude,
        border="cornflowerblue", col="cornflowerblue")

## add wells
points( x = Phelps[!vents, "Longitude"], y = Phelps[!vents, "Latitude"], panel = 1,
        pch = 2, col = "red", cex = .6)

### Add Silver Springs
points(silverPoint$Longitude, silverPoint$Latitude, col="cornflowerblue", cex = 1.3,
       lwd = 3)

key(x = par("usr")[2], y = par("usr")[4], corner = c(0,1),
    lines = list( pch = c(2, 1, 1, 1, 1),
                  col = c("red", "cornflowerblue", "lightgray",
                          "cornflowerblue", "black"),
                  lwd = c(1, 4, 2, 2, 2),
                  lty = c(rep(1, 4), 3),
                  type = c("p", "p", "l", "l", "l")),
    text = list(c("Wells", "Silver Springs", "Roads", "Rivers & Lakes",
                  "County Boundary"),
                cex = .6),
    transparent = T)

title("Kriging Predictions")

## reset par parameters
par(opar)

```