

Paper 159-2010

Exploring, Analyzing, and Summarizing Your Data: Choosing and Using the Right SAS Tool from a Rich Portfolio

Douglas Thompson, Assurant Health

ABSTRACT

This is a high-level survey of Base SAS and SAS/STAT procedures that can be used to see what your data is like, to perform analyses understandable by a non-statistician, and to summarize data. It is intended to help you to make the right choice for your specific task. Suggestions on how to make most efficient or effective use of some procedures will be offered. Examples will be provided of situations in which each procedure is likely to be useful, as well as situations in which the procedures might yield misleading results (for example, when it is better to use REG instead of CORR for looking at associations between two variables). For procedures that create listings only, or for situations in which you want to capture only part of the listing output, we will explain how to get the output with ODS. Some nifty graphical tools in PROCs UNIVARIATE and REG will be illustrated. Procedures discussed include: FREQ, SUMMARY/MEANS, UNIVARIATE, RANK, CORR, REG, CLUSTER, FASTCLUS, and VARCLUS. The intended audience for this tutorial includes all levels of SAS experience.

INTRODUCTION

All data analysts do some form of exploratory data analysis. “Exploratory data analysis” is here used to refer to basic analyses to summarize the data elements (variables), to summarize relationships among different data elements, and to simplify the data whenever it is possible to do so without losing important information. Exploratory data analyses do not require any pre-existing ideas about the data (hence the “exploratory”), although hypotheses can be used to guide parts of the data analysis. Exploratory data analysis can involve very advanced analytics techniques, but this paper focuses on relatively simple techniques. All of the techniques discussed in this paper are ones that the author uses on a regular basis in project work.

SAS provides a variety of excellent tools for exploratory data analysis. The purpose of this presentation is to illustrate some capabilities of SAS for exploratory data analysis. This is meant to provide some key illustrations. It is not meant to be exhaustive. SAS has many tools for exploratory data analysis that are not discussed in this paper.

Basic tasks in exploratory data analysis include the following:

1. *Describe the distribution of individual variables*

Examples: Describe the percentage of individuals who use different modes of transportation to work (car, bus, walk, etc.) in a dataset; describe the distribution of annual wages/salary in a dataset (average, min, max, etc.).

Useful SAS PROCs: FREQ; MEANS; UNIVARIATE

2. *Summarize the association between two variables (continuous, categorical, or one of each)*

Examples: Summarize the association between hours worked per week and age; summarize the association between annual income and hours worked per week; summarize the association between mode of transportation to work and commuting time.

Useful SAS PROCs: FREQ; REG; CORR; UNIVARIATE (with CLASS and HISTOGRAM statements)

3. *Identify variables that are redundant within a large set of variables*

Example: Identify a small set of unique variables among a large set of variables from government or industry surveys.

Useful SAS PROCs: VARCLUS; FACTOR.

4. *Group observations that are similar across a set of variables*

Example: Group counties that have a similar profile across a set of macroeconomic variables.

Useful SAS PROCs: CLUSTER; FASTCLUS.

METHODS

The exploratory data analysis techniques discussed in this paper are illustrated using the following datasets:

1. Public Use Microdata Sample (PUMS) from the 2008 American Community Survey (ACS) conducted by the U.S. Census Bureau, describing Wisconsin residents. ACS PUMS data are available for all states, but we limited the analyses to Wisconsin residents for purposes of this paper. Below, this dataset is called WI.PSAM_P55.
2. Data on Wisconsin counties (summarized at the county level) from the American Community Survey (2006-2008) and Bureau of Labor Statistics. Data from small counties are excluded. Below, this dataset is called WI.WI_COUNTIES.

Analysis techniques

For describing the distribution of individual variables, the ideal technique depends on the type of variable. For categorical variables, frequency tables are useful (PROC FREQ). For continuous variables, the distribution can be summarized using measures such as the mean, percentiles (for example, the median or 50th percentile, 25th percentile, 75th percentile), standard deviation and range (minimum and maximum values). These are available from PROCs UNIVARIATE and MEANS.

For summarizing the relationship between two variables, again the ideal technique depends on the nature of the variables. For two categorical variables, PROC FREQ provides useful tools to summarize the association. For two continuous variables, PROC CORR may be useful. However, if the relationship is not linear, PROC REG may provide a better summary of the association. Using RANK in conjunction with MEANS or UNIVARIATE is a good way to get a sense of whether or not the association is approximately linear. If it is not linear, REG provides a way to summarize the association. For pairs of variables where one is continuous and the other is categorical, UNIVARIATE with CLASS and HISTOGRAM statements can provide a good preliminary idea of the relationship. The relationship can be further summarized with REG or other modeling tools available in SAS.

For identifying redundant variables in a large set of variables, VARCLUS is a very flexible tool. VARCLUS groups of associated variables, and then the analyst can decide which variable(s) best represent each group. Principal components analysis (implemented using PROC FACTOR or PRINCOMP) provides another approach to accomplish this goal. An advantage of principal components analysis is that it provides a way of summarizing a group of variables without deleting or dropping any variables.

For grouping observations that show a similar pattern across a set of variables (for example, grouping similar states, counties, or individual people), useful tools include PROCs CLUSTER and FASTCLUS.

RESULTS

In this section, we present illustrations of some useful exploratory data analysis techniques in SAS. We also discuss situations in which certain techniques might not work well, as well as possible solutions that analysts can use in such situations.

The results are organized in terms of four basic goals of exploratory data analysis:

1. Describe the distribution of individual variables
2. Summarize the association between two variables (continuous, categorical, or one of each)
3. Identify redundant variables
4. Group observations that are similar on a set of variables

1. Describe the distribution of individual variables

To describe individual variables, the type of variable (continuous or categorical) partly determines the technique(s) that will be most useful. PROC FREQ is a very useful for summarizing the distribution of categorical variables, i.e., variables that take on a set of discrete values which essentially serve as labels, while UNIVARIATE and MEANS are useful for describing the distribution of continuous variables, i.e., variables that can take on any value within a specific range and have a meaningful relative magnitude (e.g., two is twice the magnitude of 1).

Type of worker (COW), mode of transportation to work (JWTR), health insurance (HICOV) and gender (SEX) are examples of categorical variables in the Wisconsin microdata sample from the Census Bureau. They can be summarized using PROC FREQ:

```
ods rtf file='C:\projects\MWSUG10\PROCs\PROC FREQ individual variables.rtf';
proc freq data=wi.psam_p55;
tables cow JWTR SEX HICOV;
format cow $cowf. JWTR $jwtrf. SEX $sexf. HICOV $hicovf.;
run;
ods rtf close;
```

Selected PROC FREQ output is shown below. ODS RTF produces a file with tables that can easily be copy/pasted into word processing software such as Microsoft Word. PROC FREQ produces a frequency table showing the frequency (count) and percent for each value of the variables listed in the TABLES statement. A separate table is produced for each variable. The count of observations with no data for each variable is also shown (“Frequency Missing =...”).

Transportation to work				
JWTR	Frequency	Percent	Cumulative Frequency	Cumulative Percent
Car, truck, or van	26262	88.84	26262	88.84
Bus or trolley bus	358	1.21	26620	90.05
Streetcar or trolley car	1	0.00	26621	90.05
Subway or elevated	2	0.01	26623	90.06
Railroad	9	0.03	26632	90.09
Ferryboat	1	0.00	26633	90.09
Taxicab	17	0.06	26650	90.15
Motorcycle	159	0.54	26809	90.69
Bicycle	177	0.60	26986	91.29
Walked	1009	3.41	27995	94.70
Worked at home	1401	4.74	29396	99.44
Other method	166	0.56	29562	100.00

Frequency Missing = 28642

Using PROC MEANS and/or UNIVARIATE, one can summarize the distribution of continuous variables, such as commuting time (minutes of travel time to work) or annual wages/salary. PROC MEANS is a good tool for creating a table with basic descriptive statistics, including number of observations with

non-missing values (labeled “N”), mean, minimum and maximum. This is useful for getting a basic description of variables as well as for identifying data quality issues (for example, minimum and/or maximum values that are outside of the expected range of a variable).

Example code:

```
ods rtf file='C:\projects\MWSUG10\PROCs\PROC MEANS individual variables.rtf';
proc means data=wi.psam_p55;
var WAGP JWMNP;
run;
ods rtf close;
```

Selected output from PROC MEANS:

Variable	Label	N	Mean	Std Dev	Minimum	Maximum
WAGP	PUMS Wages/salary income	47510	23589.55	36001.04	0	354000.00
JWMNP	PUMS Minutes to work	28161	22.9175455	22.2380862	1.0000000	182.0000000

PROC UNIVARIATE provides some of the same information as PROC MEANS, but many additional pieces of information are provided, for example, skewness of the distribution, percentiles, and the 5 highest and lowest values.

Example code:

```
ods rtf file='C:\projects\MWSUG10\PROCs\PROC UNIVARIATE individual
variables.rtf';
proc univariate data=wi.psam_p55 plot;
var JWMNP;
run;
ods rtf close;
```

Selected output of PROC UNIVARIATE is shown below. This provides a much more detailed picture of the distribution of an individual variable. The PLOT option provides a bar chart (histogram) of the number of values at different points along the range of a variable.

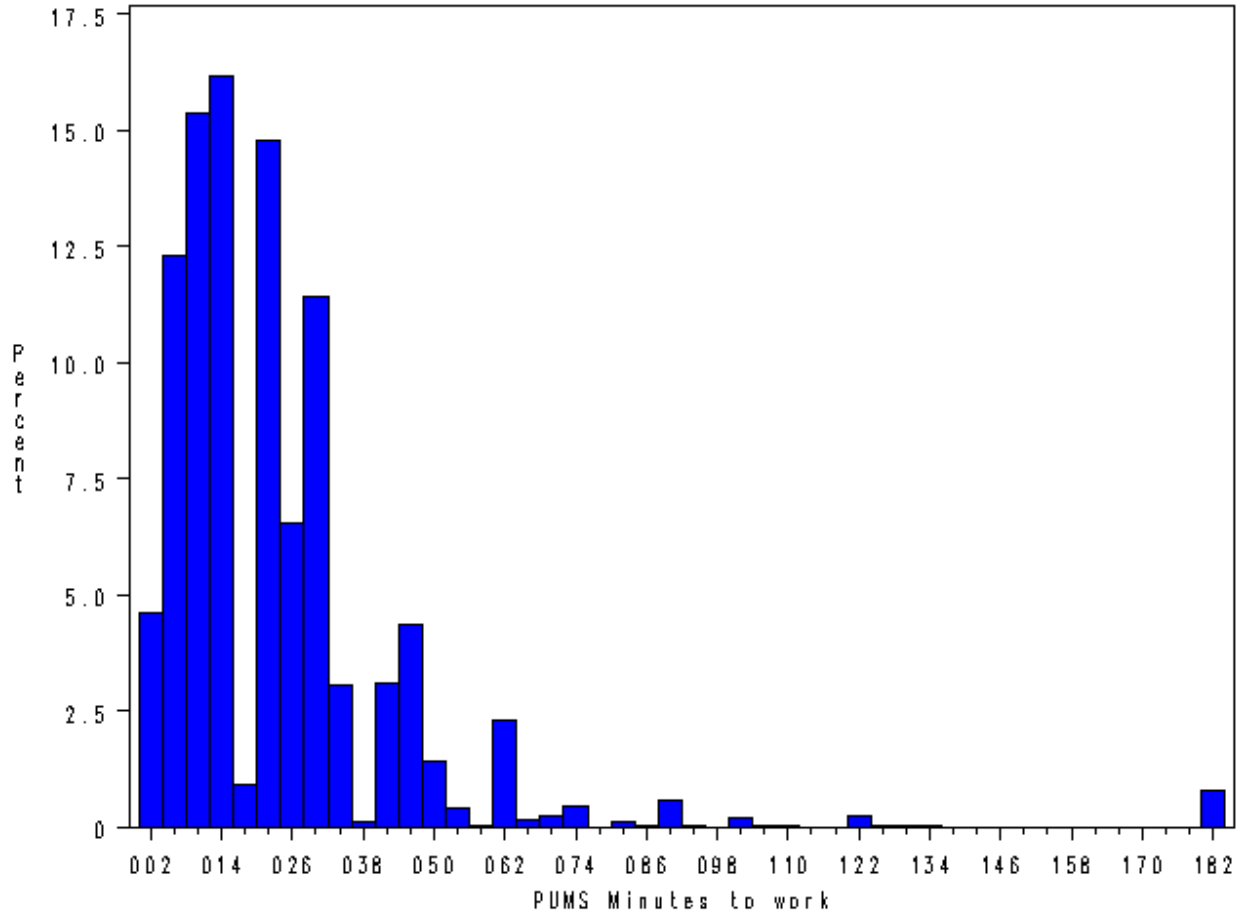
Moments			
N	28161	Sum Weights	28161
Mean	22.9175455	Sum Observations	645381
Std Deviation	22.2380862	Variance	494.532477
Skewness	3.76298342	Kurtosis	21.281249

Moments			
Uncorrected SS	28716583	Corrected SS	13926034.5
Coeff Variation	97.0352001	Std Error Mean	0.13251754

Quantiles (Definition 5)	
Quantile	Estimate
100% Max	182
99%	120
95%	60
90%	45
75% Q3	30
50% Median	20
25% Q1	10
10%	5
5%	4
1%	1
0% Min	1

Extreme Observations			
Lowest		Highest	
Value	Obs	Value	Obs
1	58127	182	57207
1	58126	182	57552
1	58107	182	57887
1	57930	182	57963
1	57888	182	58134

The HISTOGRAM statement produces a frequency histogram. This is similar to the bar chart produced with the PLOT option. However, the HISTOGRAM statement gives the analyst more control over the plot via a variety of options. Also, a set of histograms can be created by using both the CLASS and HISTOGRAM statements (as illustrated later in this paper); this can provide useful insights into the data.



2. Summarize the association between two variables (continuous, categorical, or one of each)

When looking at relationships among a small number of categorical variables (say, 2 or 3), PROC FREQ is a useful technique. Beyond 3 or so variables, the relationships tend to become complex and difficult to interpret, so other tools such as PROC LOGISTIC or classification trees might be more useful for describing the associations (these techniques are beyond the scope of this paper).

As an illustration, we can use PROC FREQ to look at associations between type of worker, health insurance coverage and gender in the Wisconsin microdata sample from the Census Bureau.

Example code:

```
ods rtf file='C:\projects\MWSUG10\PROCs\cow by HICOV.rtf';
proc freq data=wi.psam_p55;
tables cow*HICOV / out=a outpct;
```



```
format cow $cowf. HICOV $hicovf.;  
run;  
ods rtf close;
```

Selected output of PROC FREQ is shown below. The count of individuals with vs. without health insurance (columns) is shown for each type of worker (rows). A set of percentages is provided for each cell. Different percentages are useful for different purposes. For example, the row percentages (3rd row in each cell) add to 100; they show the percentage of workers of each type who have health insurance (e.g. , 89.65% of workers for private for-profit companies have health insurance, compared with 94.73% who work for non-profit organizations).

Table of COW by HICOV			
COW(Class of worker)	HICOV(Any health insurance coverage)		
	With health insurance coverage	No health insurance coverage	Total
Frequency Percent Row Pct Col Pct			
Employee of a private for-profit company	22346 60.44 89.65 66.87	2580 6.98 10.35 72.61	24926 67.42
Employee of a private not-for-profit organization	2895 7.83 94.73 8.66	161 0.44 5.27 4.53	3056 8.27
Local government employee (city, county, etc.)	2679 7.25 96.78 8.02	89 0.24 3.22 2.50	2768 7.49
State government employee	1229 3.32 95.35 3.68	60 0.16 4.65 1.69	1289 3.49
Federal government employee	533 1.44 95.69 1.59	24 0.06 4.31 0.68	557 1.51
Self-employed in own not incorporated business	2323 6.28 83.44 6.95	461 1.25 16.56 12.97	2784 7.53
Self-employed in own incorporated business	1190 3.22 91.61 3.56	109 0.29 8.39 3.07	1299 3.51
Working without pay in family business or farm	124 0.34 79.49 0.37	32 0.09 20.51 0.90	156 0.42
Unemployed	100 0.27 72.99 0.30	37 0.10 27.01 1.04	137 0.37

Table of COW by HICOV			
COW(Class of worker)	HICOV(Any health insurance coverage)		
	With health insurance coverage	No health insurance coverage	Total
Frequency			
Percent			
Row Pct			
Col Pct			
Total	33419 90.39	3553 9.61	36972 100.00
Frequency Missing = 21232			

The OUTPUT statement produces the same information, but in the form of a SAS dataset that can easily be exported to other software for display or analysis. The dataset produced by OUTPUT can be further manipulated in SAS; for example, sub-analyses could be done for particular types of workers.

COW	HICOV	COUNT	PERCENT	PCT_ROW	PCT_COL
	With health insurance coverage	20189			
	No health insurance coverage	1043			
Employee of a private for-profit company	With health insurance coverage	22346	60.4403332	89.6493621	66.866154
Employee of a private for-profit company	No health insurance coverage	2580	6.97825381	10.3506379	72.6146918
Employee of a private not-for-profit organization	With health insurance coverage	2895	7.83024992	94.7316754	8.66273677
Employee of a private not-for-profit organization	No health insurance coverage	161	0.43546468	5.26832461	4.53138193
Local government employee (city, county, etc.)	With health insurance coverage	2679	7.24602402	96.7846821	8.01639786
Local government employee (city, county, etc.)	No health insurance coverage	89	0.24072271	3.21531792	2.50492542
State government employee	With health insurance coverage	1229	3.32413718	95.3452289	3.6775487
State government employee	No health insurance coverage	60	0.16228497	4.65477114	1.68871376
Federal government employee	With health insurance coverage	533	1.4416315	95.6912029	1.5949011
Federal government employee	No health insurance coverage	24	0.06491399	4.30879713	0.67548551
Self-employed in own not incorporated business	With health insurance coverage	2323	6.28313318	83.441092	6.95113558
Self-employed in own not incorporated business	No health insurance coverage	461	1.24688954	16.558908	12.9749507
Self-employed in own incorporated business	With health insurance coverage	1190	3.21865195	91.6089299	3.56084862
Self-employed in own incorporated business	No health insurance coverage	109	0.2948177	8.39107005	3.06783
Working without pay in family business or farm	With health insurance coverage	124	0.33538894	79.4871795	0.37104641
Working without pay in family business or farm	No health insurance coverage	32	0.08655199	20.5128205	0.90064734
Unemployed	With health insurance coverage	100	0.27047495	72.9927007	0.29923098
Unemployed	No health insurance coverage	37	0.10007573	27.0072993	1.04137349

PROC FREQ is not limited to 2-way tables (i.e., TABLES A * B). Frequency tables can be done with 3 or more variables. If 3 variables are used in a TABLES statement (TABLES A * B * C), a separate table of B by C is produced for each level of A. For example:

```
ods rtf file='C:\projects\MWSUG10\PROCs\sex by cow by HICOV.rtf';
proc freq data=wi.psam_p55;
tables sex*cow*HICOV / out=a outpct;
format cow $cowf. HICOV $hicovf. sex $sexf.;
run;
ods rtf close;
```

The output will be separate table for each gender category, showing type of worker by health insurance coverage (the output is not shown in this paper).

When one wants to look at the relationship between two variables, where one variable is categorical and the other is continuous, PROC MEANS with a CLASS statement may provide a good summary. For example, we can look at the association between mode of transportation to work (JWTR) and commuting time (daily travel time to work in minutes, JWMNP) using the following code:

```
ods rtf file='C:\projects\MWSUG10\PROCs\mode of transport by commuting
time.rtf';
proc means data=wi.psam_p55;
class JWTR;
var JWMNP;
format JWTR $jwtrf.;
run;
ods rtf close;
```

Selected output:

Analysis Variable : JWMNP PUMS Minutes to work						
Transportation to work	N Obs	N	Mean	Std Dev	Minimum	Maximum
Car, truck, or van	26262	26262	23.2481913	21.9242235	1.0000000	182.0000000
Bus or trolley bus	358	358	38.9217877	23.8199249	1.0000000	182.0000000
Streetcar or trolley car	1	1	20.0000000	.	20.0000000	20.0000000
Subway or elevated	2	2	25.0000000	7.0710678	20.0000000	30.0000000
Railroad	9	9	114.8888889	47.5695397	40.0000000	182.0000000
Ferryboat	1	1	5.0000000	.	5.0000000	5.0000000
Taxicab	17	17	17.8823529	13.1428800	1.0000000	60.0000000
Motorcycle	159	159	21.3459119	19.4122457	1.0000000	100.0000000
Bicycle	177	177	16.3954802	18.4317408	1.0000000	182.0000000
Walked	1009	1009	7.8572844	12.0458552	1.0000000	182.0000000
Worked at home	1401	0
Other method	166	166	31.7228916	48.7999785	1.0000000	182.0000000

This shows that among the more common modes of transportation (say, those with a frequency of 100 or more), bus riders have the longest average commutes (about 39 minutes) while walkers have the shortest average commutes (about 8 minutes).

Other techniques are useful for summarizing relationships between two continuous variables. An example is the relationship between a person's age (AGEP) and average hours worked per week (WKHP) from the Wisconsin microdata sample provided by the Census Bureau. PROC CORR indicates a significant, positive association between these two variables (i.e., as one increases, the other increases), as indicated by the positive sign of the coefficient (0.09916) and p-value less than 0.05 (a traditional cutpoint of statistical significance). The code to execute PROC CORR is:

```
ods rtf;
proc corr data=wi.psam_p55 pearson spearman;
var WKHP AGEP;
run;
```

Selected output from PROC CORR:

Pearson Correlation Coefficients Prob > r under H0: Rho=0 Number of Observations		
	WKHP	AGEP
WKHP PUMS Hours worked per week	1.00000 33574	0.09916 <.0001 33574
AGEP PUMS Age	0.09916 <.0001 33574	1.00000 58204

The table above shows the Pearson coefficient. The Spearman coefficient (which correlates ranks of the two variables) pointed to the same conclusion.

However, graphical inspection shows that the positive relationship indicated by PROC CORR is not realistic. This was discovered by deciling hours worked per week (i.e., creating 10 approximately equal-sized, ranked groups) using PROC RANK, then the mean and median of age was computed for each of the 10 groups. The code to do this is illustrated below.

```
proc rank data=wi.psam_p55(where=(WKHP ne .)) out=rank_wi_pums groups=10;
var AGEP;
ranks rank_AGEP;
run;

proc univariate data=rank_wi_pums noprint;
class rank_AGEP;
var WKHP;
```

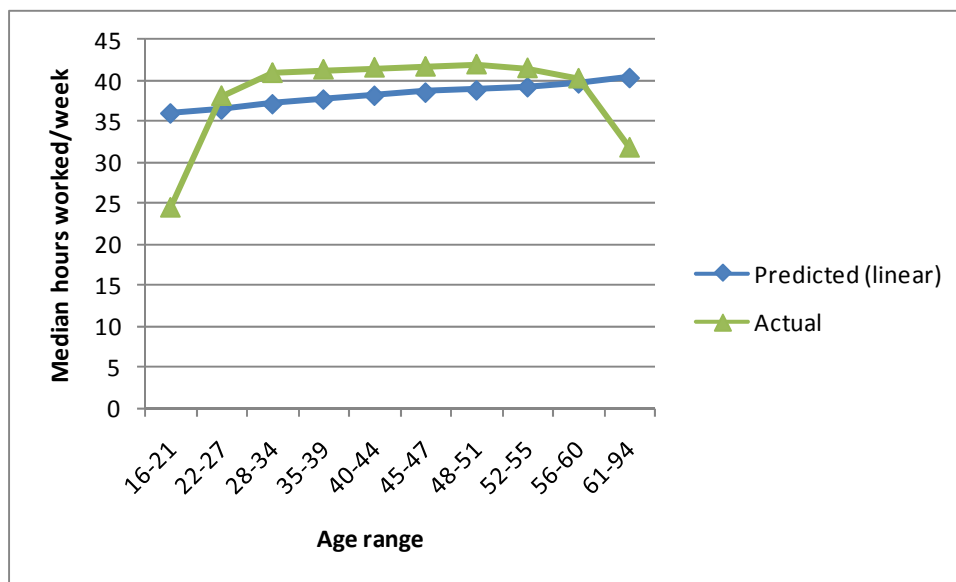
```

output out=a mean=mean;
run;
proc univariate data=rank_wi_pums noprint;
class rank_AGEP;
var AGE;
output out=b min=min_age max=max_age median=median_age;
run;

data c;
merge a b;
by rank_AGEP;
run;

```

The results (SAS dataset C) were graphed. The estimated linear slope was also plotted on the same graph (the linear equation is: $\text{hours worked per week} = 34.22843 + \text{age} * 0.09476$; code to estimate this equation is shown below). The graph is shown below. The actual relationship has an inverse U-shape. The linear slope (which is basically the representation used by PROC CORR) does not provide a good summary of the relationship.



When there is a non-linear relationship between two continuous variables, polynomial terms can sometimes provide a good summary of the relationship. For example, if the relationship has a U-shape (or inverse U-shape, as seems to be the case for age vs. hours worked per week), a quadratic model can provide a good summary of the relationship. PROC REG was used to estimate linear and quadratic models of the association between age and hours worked per week. The code and results are shown below.

```

* get the linear and quadratic equations using the entire dataset;
data wi_pums;
set wi.psam_p55;
AGEP_sq=AGEP**2;

```

```
run;

* linear equation;
ods rtf;
proc reg data=wi_pums;
model WKHP = AGEP;
run;
quit;
```

Selected output of PROC REG for the linear equation:

Parameter Estimates						
Variable	Label	DF	Parameter Estimate	Standard Error	t Value	Pr > t
Intercept	Intercept	1	34.22843	0.23447	145.98	<.0001
AGEP	PUMS Age	1	0.09476	0.00519	18.26	<.0001

```
* quadratic equation;
ods rtf;
proc reg data=wi_pums;
model WKHP = AGEP AGEP_sq;
run;
quit;
```

Selected output of PROC REG for the quadratic equation:

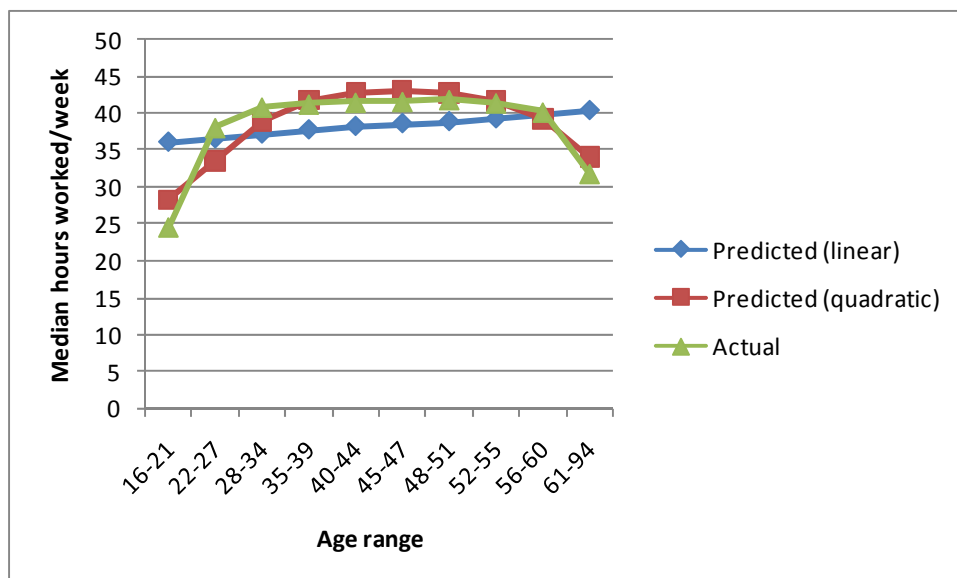
Parameter Estimates						
Variable	Label	DF	Parameter Estimate	Standard Error	t Value	Pr > t
Intercept	Intercept	1	-1.27825	0.51385	-2.49	0.0129
AGEP	PUMS Age	1	1.97777	0.02517	78.56	<.0001
AGEP_sq		1	-0.02207	0.00028960	-76.19	<.0001

The two equations can be summarized as follows:

Linear: Predicted hours worked per week = $34.22843 + \text{age} * 0.09476$

Quadratic: Predicted hours worked per week = $-1.27825 + \text{age} * 1.97777 + \text{age}^2 * -0.02207$

The two equations are plotted against the actual values in the graph below. The quadratic equation is a closer fit to the actual values than is the linear equation.



This example shows how the results of PROC CORR can be misleading. The summary of the relationship between age and hours worked from PROC CORR (as well as REG with a linear equation) was misleading. However, the summary from PROC REG with a quadratic term seemed to provide a fairly good summary of the relationship between age and average hours worked per week – it captures the inverse U-shaped association.

Using the same example data, PROC UNIVARIATE provides another useful tool for summarizing the relationship between age and hours worked per week. The HISTOGRAM statement in PROC UNIVARIATE allows one to take a more detailed look at the curvilinear (specifically, inverse U-shaped) relationship shown in the figures above. First, age is grouped into categories -- below we use three categories of age, but one could use any number of categories that seem useful. Then a histogram of average hours worked per week is created for each age category using PROC UNIVARIATE. Code to do this is shown below.

```

data wi_pums2;
set wi.psam_p55;
if AGEP ne . then do;
    if AGEP<=22 then age_cat=1;
    else if AGEP<=65 then age_cat=2;
    else age_cat=3;
end;
run;

proc format;
value agecf 1='Age<=22' 2='Age 23-65' 3='Age>65';
run;

ods html;
proc univariate data=wi_pums2 noprint;
format age_cat agecf.;
class age_cat;

```

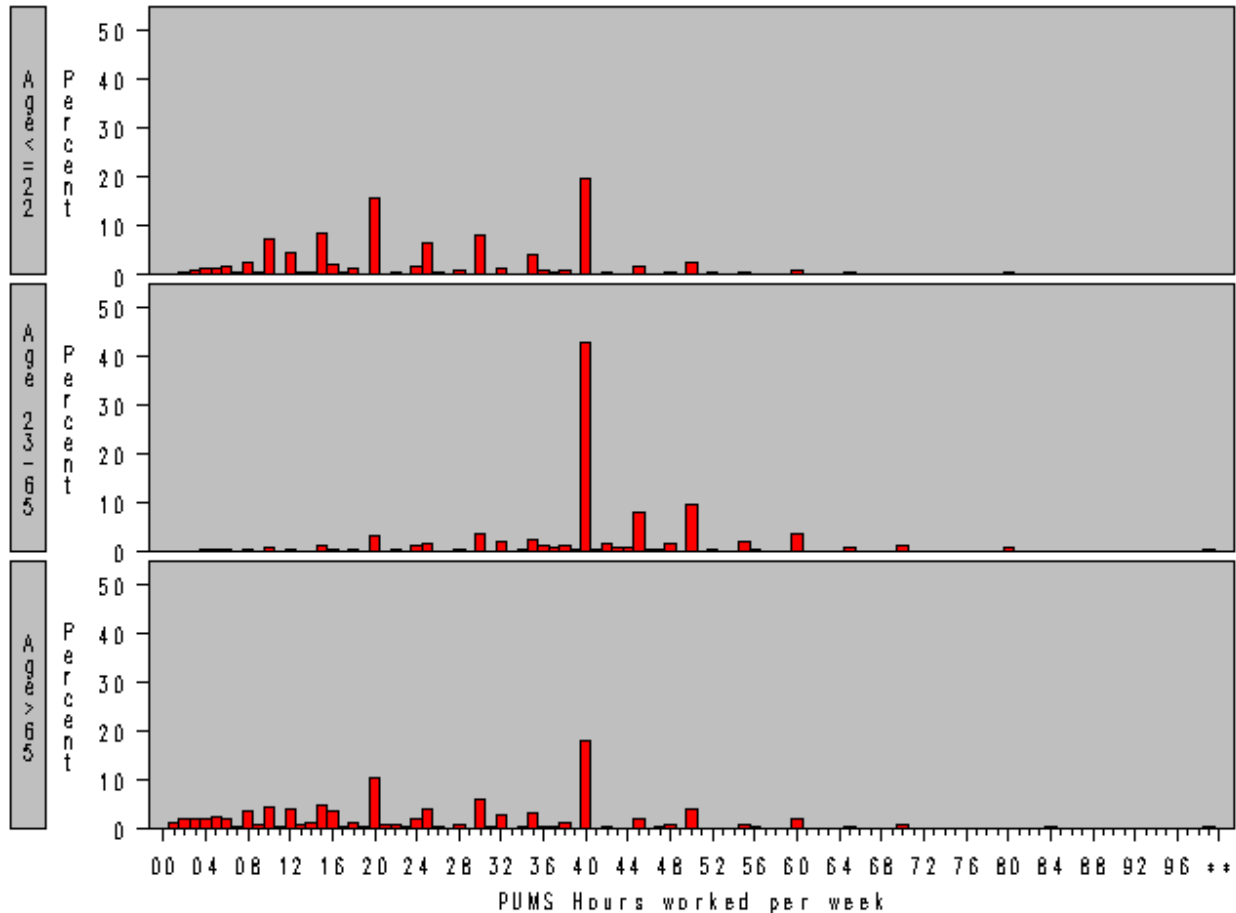


```

    histogram WKHP / nrows      = 3
                                cframe   = ligr
                                cfill    = red
                                cframeside = ligr
midpoints = 0 to 100 by 1;
run;

```

Selected output produced by the code above:



The histograms show that the Wisconsin residents who worked less than 40 hours/week were more numerous in the age groups less than 23 years old and greater than 65 years old, while the age 23-65 group had the greatest percentage of individuals working 40 or more hours per week.

A common problem in exploratory data analysis occurs when one attempts to summarize a relationship between two variables in a sample that includes a mixture of different populations. One way to put it is, when one has a sample of apples and oranges, it is often best to separate the apples from the oranges before trying to summarize a relationship in the sample – the relationship may be very different for the apples and oranges, and the relationship estimated in the mixed sample has ambiguous meaning. Graphics in PROC REG are useful for identifying such situations. An example is the association of annual wages with hours worked per week in the Wisconsin microdata sample from the Census Bureau.

Logically, one might think that as hours worked per week increases, annual wages would also increase. This relationship was estimated in PROC REG using the following code:

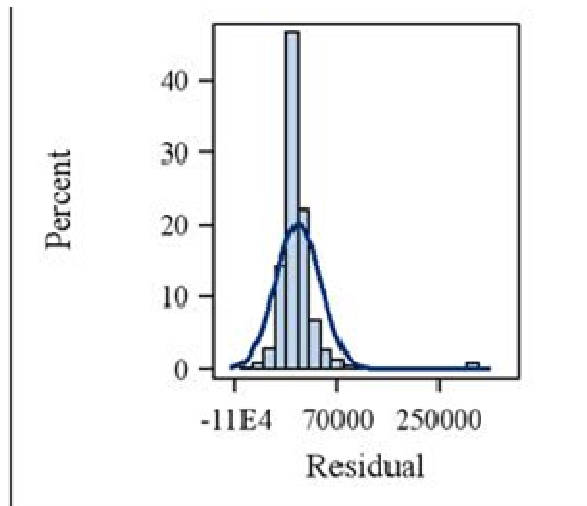
```
data wi_pums;
set wi.psam_p55;
rand=ranuni(26372);
run;

proc sort data=wi_pums;
by rand;
run;

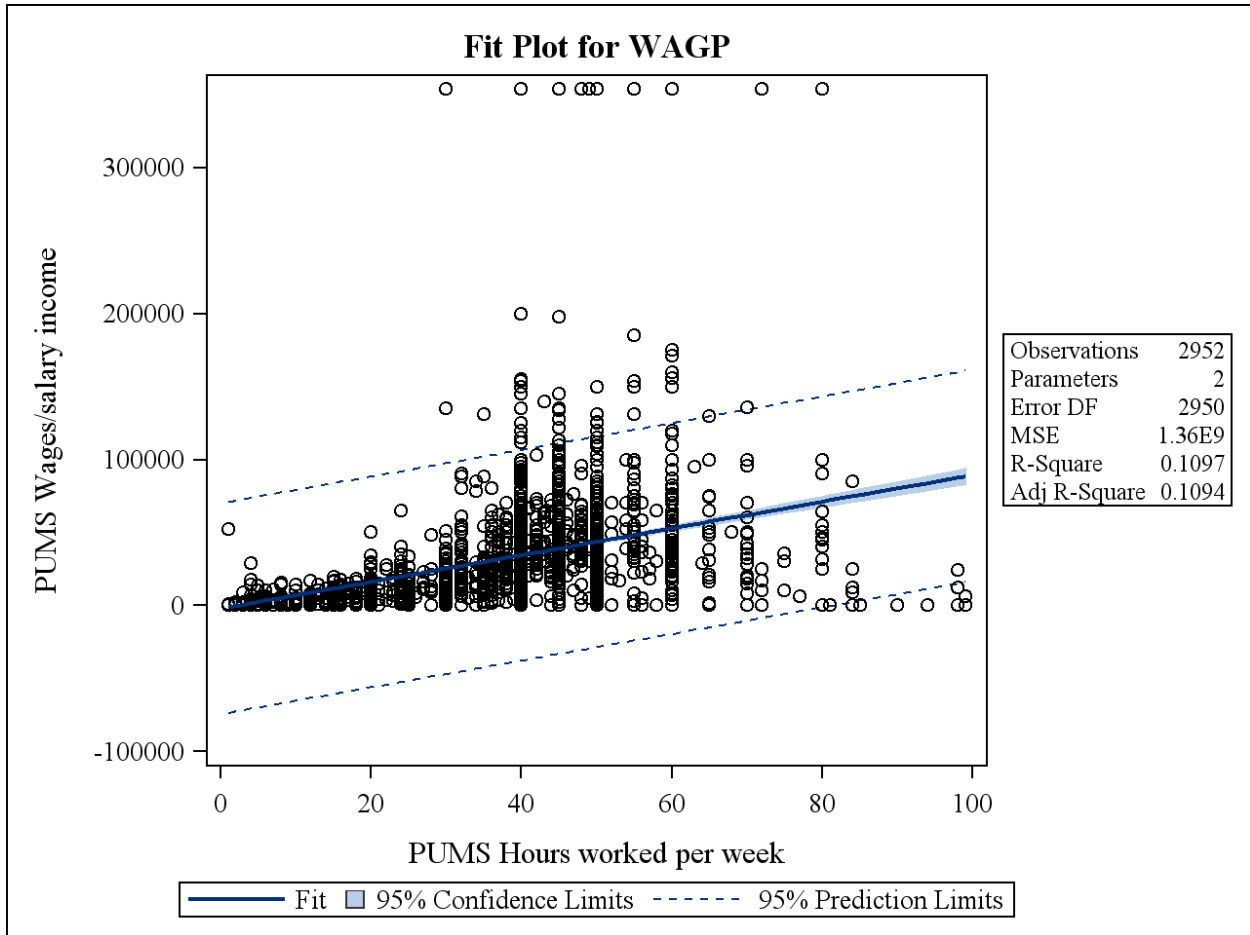
data wi_pums_samp;
set wi_pums;
if _n_ <= 5000;
run;

ods rtf;
ods graphics on;
proc reg data=wi_pums_samp;
model WAGP = WKHP;
run;
quit;
ods graphics off;
```

There was some support for the initial hypothesis. The estimated relationship was positive and significant: the model indicated that salary increases by about \$916 for each additional hour worked per week. However, the r-squared value (0.11) was surprisingly small. This suggests that hours worked per week explains only about 11% of the variance in annual income; one might expect it to be more. The output of ODS GRAPHICS quickly points to a possible problem. The residual distribution is non-normal – there are some extremely high residuals. Specifically, the model over-predicts some individuals' annual wages by more than \$250K. See the residual plot below. This probably means that some people are working long hours, but they are getting a very small wage.



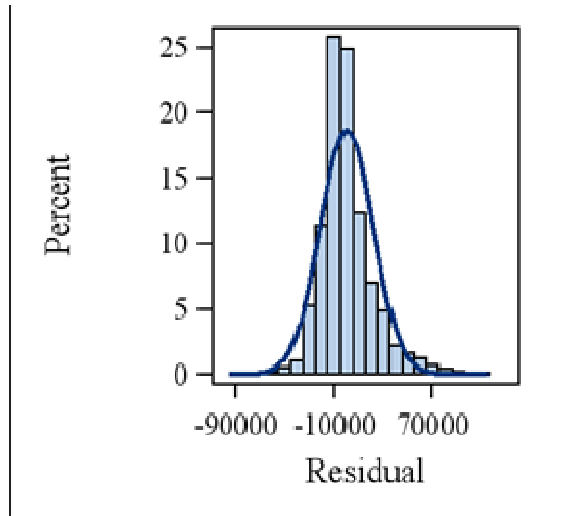
Inspection of a scatterplot of the two variables (produced by ODS GRAPHICS in PROC REG) quickly points to a likely source of the problem (see scatterplot below). First, there is a group of very high wage earners whose earnings appear to be unrelated to the hours that they work per week. Second, there seems to be an even larger group of individuals who earn very little or no wages – some of these individuals work very long hours (maybe they are dedicated volunteers, or they get paid in a form other than wages, such as stock options). These were not the populations that we had in mind when we started this analysis. These are the oranges, and we need to separate them from the apples before we try to estimate the relationship of interest.



The model is refitted after removing the individuals who have very high wages or zero wages, using the following code.

```
ods rtf;
ods graphics on;
proc reg data=wi_pums_samp (where=(0<WAGP<150000));
model WAGP = WKHP;
run;
quit;
ods graphics off;
```

This gives us a much better-fitting model. The coefficient increases only slightly (\$1,009 additional annual wages for each additional hour worked per week). However, other indicators of model fit improve markedly – for example, r-squared nearly triples to 0.28 and the residual distribution is much closer to normal, as indicated by the residual plot shown below.



The model is far from perfect, but it is much improved. This shows the importance of sifting diverse samples before trying to summarize a relationship between two variables. There are many other useful features of ODS GRAPHICS for PROC REG that are also helpful (for example, the Cook's d chart for identifying observations that may have an excessive influence on the model) but these are beyond the scope of this paper.

3. Identify redundant variables

Government and marketing databases (e.g., Census, Centers for Disease Control and Prevention, Bureau of Labor Statistics, Acxiom, Claritas, KBM, etc.) typically contain hundreds or even thousands or variables. Analyzing all of the variables in a dataset may be neither feasible nor necessary, especially given that many of the variables may be basically redundant (that is, they are closely related to each other and contain similar information). A common task in exploratory data analysis is to take a large set of variables and reduce it to a smaller set of variables to focus on in subsequent analyses.

PROC VARCLUS and FACTOR (with METHOD=PRIN for principal components analysis) can be used to identify sets of variables that contain similar information, i.e., they have some degree of redundancy. Both PROCs utilize principal components analysis (PCA). PCA is a method for identifying a limited set of components (a weighted linear combinations of the observed variables) that are uncorrelated with each other, but together they explain a maximal percentage of the total variance in the observed variables (for a good, intuitive explanation of what this means, see Hatcher, 1994). The components can be transformed in various ways or “rotated” to make them more easy to interpret. Varimax-rotated components (ROTATE=VARIMAX option in PROC FACTOR with METHOD=PRIN) tend to have high

loadings (correlations) with distinct sets of variables. A set of variables that together have high loadings on a given component are inter-correlated and contain similar information.

Below is illustrative PCA code using the Wisconsin microdata sample from the Census Bureau:

```
ods rtf;
proc factor data=wi.psam_p55 method=prin rotate=varimax scree;
var AGEP INTIP JWMNP JWRIP MARHYP OIP PAP RETP SEMP SSIP SSP WAGP WKHP PERNP
PINCP POVPIP;
run;
```

Using VARIMAX rotation (ROTATE=VARIMAX), typically different variables will have high correlations with different components. The scree plot (SCREE option) is helpful in deciding how many components are meaningful. Variables with loadings of 0.40 or more in absolute value (positive or negative) with a given component are the ones most strongly correlated with that component. Variables with loadings of 0.40 or more in absolute value are bolded in the table below (selected output from PROC FACTOR).

Rotated Factor Pattern					
		Factor1	Factor2	Factor3	Factor4
AGEP	PUMS Age	0.87806	0.12633	-0.07072	0.08966
JWMNP	PUMS Minutes to work	-0.02176	0.13698	0.71933	-0.04203
JWRIP	PUMS Total riders	-0.00230	-0.10220	0.73834	0.01035
MARHYP	PUMS Year last married	-0.84284	-0.08272	0.08158	-0.08903
PAP	PUMS SSI/AFDC/other welfare income	-0.00717	-0.09377	-0.03056	0.11009
RETP	PUMS Retirement income	0.39698	-0.03891	0.06248	-0.20146
SEMP	PUMS Self-employment income	0.02374	0.03941	0.04492	0.93190
SSIP	PUMS Social Security or Railroad Retirement Income	0.65866	-0.16949	0.02779	-0.01096
WAGP	PUMS Wages/salary income	-0.07638	0.80293	-0.00145	-0.17357
WKHP	PUMS Hours worked per week	-0.33569	0.60282	0.04990	0.27114
POVPIP	PUMS Poverty index	0.19091	0.72921	-0.02149	-0.02118

The scree plot produced by PROC FACTOR indicates that after the first 2 or 3 components (“Factor1,” “Factor2,” “Factor3”), not much information is yielded by the other components. Therefore, we might focus on the first 3 components to identify correlated/redundant variables. There is an art to interpreting the results of PCA and it requires some judgment from the analyst. In this example, it looks like the first component represents age (high loadings for age, year last married and various forms of retirement income – these variables are inter-correlated and contain some similar information). The second component seems to represent total household income. The third component has to do with transportation to work.

There are several ways that one could use this information. One is to create a scale or component score, combining several variables to create a composite variable. For example, one could devise a scale where one adds together the variables that have a high loading on a given component. Or a weighted combination could be used, such as a component score (see Hatcher, 1994, for more details on that). A third strategy is to simply pick one variable to represent the whole component. With this last strategy, some information is lost, but that might not have a major impact on the resulting analysis.

PROC VARCLUS is an alternative technique to identify redundant variables. It is related to PCA (VARCLUS uses an iterative algorithm that is applied to a PCA), but it places variables into distinct clusters and typically the analyst chooses one variable in the cluster to represent the entire cluster (or more than one variable if the variables in the cluster are not closely inter-correlated).

Code to implement PROC VARCLUS using the Wisconsin microdata sample from the Census Bureau is shown below:

```
ods rtf;
proc varclus data=wi.psam_p55;
var AGEP JWMNP JWRIP MARHYP PAP RETP SEMP SSP WAGP WKHP POVPIP;
run;
```

Selected output from PROC VARCLUS:

4 Clusters		R-squared with			Variable Label
Cluster	Variable	Own Cluster	Next Closest	1-R**2 Ratio	
Cluster 1	AGEP	0.8058	0.0034	0.1949	PUMS Age
	MARHYP	0.7507	0.0037	0.2502	PUMS Year last married
	RETP	0.1417	0.0021	0.8600	PUMS Retirement income
	SSP	0.4282	0.0224	0.5849	PUMS Social Security or Railroad Retirement Income
Cluster 2	PAP	0.0071	0.0000	0.9929	PUMS SSI/AFDC/other welfare income
	WAGP	0.6608	0.0027	0.3401	PUMS Wages/salary income
	WKHP	0.4298	0.0492	0.5998	PUMS Hours worked per week
	POVPIP	0.4999	0.0166	0.5086	PUMS Poverty index
Cluster 3	JWMNP	0.5386	0.0052	0.4638	PUMS Minutes to work
	JWRIP	0.5386	0.0014	0.4621	PUMS Total riders
Cluster 4	SEMP	1.0000	0.0023	0.0000	PUMS Self-employment income

This analysis shows that there are 4 clusters of variables. It is useful to look at each variable's r-squared with its own cluster ("R-squared with Own Cluster") to identify variables that are strongly related to the cluster – the higher the r-squared, the more strongly associated a given variable is with the cluster. In this example, it turns out that the variables that are in the same clusters are also the variables with high loadings on the same components in PCA (i.e., there is an age cluster, a household income cluster, and a transportation cluster). This will often be the case, although not necessarily always. One way to use the results from PROC VARCLUS is to choose one variable to represent each cluster and exclude the other variables from subsequent analyses. However, if r-squared with the cluster is low, it might make sense to retain multiple variables with the cluster for subsequent analyses.

4. *Group observations that are similar on a set of variables*

Identifying redundant variables essentially involves clustering variables. It might also be of interest to identify clustered observations (rows in a dataset), that is, observations that show a similar pattern across a set of variables. The observations (rows) might represent individual people, counties, states and so forth.

PROC CLUSTER and PROC FASTCLUS are two SAS/STAT procedures that can be used to cluster observations. PROC CLUSTER often requires slightly more work to use and will not be described in this paper, but it is one of the tools that SAS offers for grouping similar observations.

PROC FASTCLUS puts every observation into one of k clusters, where k is chosen by the analyst. Euclidean distance defines the distance from cluster centroids. There are heuristics to determine the optimal number of clusters (for example, the Pseudo-F statistic, which is output by PROC FASTCLUS by default) but it is also useful to take interpretability into account. Generally, the higher the Pseudo-F statistic, the better a given cluster solution fits the data.

When variables are on different scales, it may make sense to standardize them prior to analysis (one can easily do this using PROC STANDARD as illustrated below), otherwise some of the variables can have undue influence on the outcome.

Code to implement PROC FASTCLUS is illustrated below. This code uses data from Wisconsin counties from the American Community Survey and Bureau of Labor Statistics.

```
proc standard data=wi.wi_counties out=wi_counties mean=0 std=1;
var pct_ownocc_MrtgIncRat_gt4_08
   _HH_MRINC_0608
   pct_civpop_insured_08
   pct_hous_vacant_0608
   pct_2564_ClgGrd_08
   _Median_age_y_0608
   _ur_Ann_2008;
run;

* St Croix county has missing data, therefore drop it;
data wi_counties_no_stcroix;
```

```

set wi_counties(where=(county_state ne 'STCROIX_WI'));
run;

ods listing;
proc fastclus data=wi_counties_no_stcroix maxclusters=2 out=outclus;
var pct_ownocc_MrtgIncRat_gt4_08
_HH_MRINC_0608
pct_civpop_insured_08
pct_hous_vacant_0608
pct_2564_ClgGrd_08
_Median_age_y_0608
_ur_Ann_2008;
run;

```

Solutions with different numbers of clusters (2 to 5) were tried. A 2-cluster solution was chosen based on a combination of Pseudo-F statistics and interpretability of the resulting clusters.

Selected output from PROC FASTCLUS is shown below.

There were 18 counties in Cluster 1, and 4 counties in Cluster 2.

Cluster Summary						
Cluster	Frequency	RMS Std Deviation	Maximum Distance from Seed to Observation	Radius Exceeded	Nearest Cluster	Distance Between Cluster Centroids
1	18	0.8333	4.7150		2	3.8519
2	4	0.8655	2.6982		1	3.8519

The following table output by PROC FASTCLUS shows means of the analytic variables for each cluster.

Cluster Means			
Cluster	pct_ownocc_MrtgIncRat_gt4_08 (mortgage/income ratio > 4)	_HH_MRINC_0608 (household income)	pct_civpop_insured_08 (% w/health insurance)
1	-0.283718123	-0.460642934	-0.268104979
2	0.971307751	1.644671904	1.149336580

Cluster Means				
Cluster	pct_hous_vacant_0608 (% vacancy for housing)	pct_2564_ClgGrd_08 (% w/graduate degree)	_Median_age_y_0608 (median age)	_ur_Ann_2008 (unemployment)
1	0.085276793	-0.382292958	-0.100442664	0.289923653
2	-0.383745568	1.562215691	0.670770543	-1.201657247

Cluster means are on the standardized scale (0 is average, + is above average, and – is below average). It is clear that Cluster 2 is relatively affluent (high income and education, low unemployment and housing vacancy). Cluster 2, the more affluent one, consists of Dane County (Madison) and the Milwaukee suburbs (Ozaukee, Washington and Waukesha Counties), while Cluster 1 is the rest of the state. This makes intuitive sense.

CONCLUSION

All analysts do some form of exploratory data analysis. Common tasks at the exploratory data analysis stage include: 1) describe the distribution of individual variables; 2) summarize the relationship between two variables; 3) identify and deal with (for example, combine or delete) redundant variables within a large set of variables; and 4) group observations that are similar across a set of variables. SAS has some excellent tools for these exploratory data analysis tasks. The purpose of this presentation was to illustrate some key capabilities of SAS in this area. The goal was to hit some highlights; this paper was not meant to be exhaustive of the exploratory data analysis capabilities of SAS. We also discussed situations in which some of the techniques might not work well, along with alternative analytic approaches to deal with such situations.

REFERENCES

Hatcher L. (1994). A step-by-step approach to using the SAS system for factor analysis and structural equation modeling. Cary, NC: SAS Institute Press.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Doug Thompson
Assurant Health
500 West Michigan
Milwaukee, WI 53203
Work phone: (414) 299-7998
E-mail: Doug.Thompson@Assurant.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration. Other brand and product names are trademarks of their respective companies.