PROC步

常用通用选项

VAR:规定用这个过程分析的一些变量；

WEIGHT:规定一个变量，它的值是这些观测的相应权数；

CLASS:在分析中指定一些变量为分类变量；

BY:规定一些变量，SAS过程对输入数据集用by变量定义的几个 数据组分别进行分析处理；

OUTPUT/OUT:给出用该过程产生的输出数据集的信息；

In [1]:
```
OPTIONS COMPRESS = YES;

DATA CARS;
SET SASHELP.CARS;
RUN;
```

```
SAS Connection established. Subprocess id is 11770
```

Out[1]:

```
34   ods listing close;ods html5 (id=saspy_internal) file=stdout options(bitmap_mode='inline') devi
ce=svg style=HTMLBlue; ods
34 ! graphics on / outputfmt=png;
NOTE: Writing HTML5(SASPY_INTERNAL) Body file: STDOUT
35
36   OPTIONS COMPRESS = YES;
37
38   DATA CARS;
39   SET SASHELP.CARS;
40   RUN;
NOTE: There were 428 observations read from the data set SASHELP.CARS.
NOTE: The data set WORK.CARS has 428 observations and 15 variables.
NOTE: Compressing data set WORK.CARS decreased size by 0.00 percent.
      Compressed is 2 pages; un-compressed would require 2 pages.
NOTE: DATA statement used (Total process time):
      real time           0.00 seconds
      cpu time            0.00 seconds

41
42   ods html5 (id=saspy_internal) close;ods listing;

43
```

PROC SORT

(1)基本形式:

PROC SORT DATA = OUT = ; BY ; RUN;

SAS会按照第一个变量先排序，再对后面排序；

SAS默认排序是按升序排序，若需降序排序，在变量名前加DESCENDING；

(2)排序设置：

NODUPKEY、NOUNIQUEKEY、NODUP(探索);

In [2]:
```
/* PROC SORT */
PROC SORT DATA = SASHELP.CARS;
BY MSRP;
RUN;
```

Out[2]:
```
45   ods listing close;ods html5 (id=saspy_internal) file=stdout options(bitmap_mode='inline') devi
ce=svg style=HTMLBlue; ods
45 ! graphics on / outputfmt=png;
NOTE: Writing HTML5(SASPY_INTERNAL) Body file: STDOUT
46
47   /* PROC SORT */
48   PROC SORT DATA = SASHELP.CARS;
49   BY MSRP;
50   RUN;
ERROR: User does not have appropriate authorization level for library SASHELP.
NOTE: The SAS System stopped processing this step because of errors.
NOTE: PROCEDURE SORT used (Total process time):
      real time           0.01 seconds
      cpu time            0.00 seconds


51
52   ods html5 (id=saspy_internal) close;ods listing;

53
```

In [3]:
```
PROC SORT DATA = SASHELP.CARS OUT = CARS_MSRP;
BY DESCENDING MSRP;
RUN;

PROC SORT DATA = SASHELP.CARS OUT = CARS_MAKE;
BY MAKE;
RUN;
```

Out[3]:
```
55   ods listing close;ods html5 (id=saspy_internal) file=stdout options(bitmap_mode='inline') devi
ce=svg style=HTMLBlue; ods
55 ! graphics on / outputfmt=png;
NOTE: Writing HTML5(SASPY_INTERNAL) Body file: STDOUT
56
57   PROC SORT DATA = SASHELP.CARS OUT = CARS_MSRP;
58   BY DESCENDING MSRP;
59   RUN;
NOTE: There were 428 observations read from the data set SASHELP.CARS.
NOTE: The data set WORK.CARS_MSRP has 428 observations and 15 variables.
NOTE: Compressing data set WORK.CARS_MSRP decreased size by 0.00 percent.
      Compressed is 2 pages; un-compressed would require 2 pages.
NOTE: PROCEDURE SORT used (Total process time):
      real time           0.00 seconds
      cpu time            0.00 seconds


60
61   PROC SORT DATA = SASHELP.CARS OUT = CARS_MAKE;
62   BY MAKE;
63   RUN;
NOTE: There were 428 observations read from the data set SASHELP.CARS.
NOTE: The data set WORK.CARS_MAKE has 428 observations and 15 variables.
NOTE: Compressing data set WORK.CARS_MAKE decreased size by 0.00 percent.
      Compressed is 2 pages; un-compressed would require 2 pages.
NOTE: PROCEDURE SORT used (Total process time):
      real time           0.00 seconds
      cpu time            0.00 seconds


64
65   ods html5 (id=saspy_internal) close;ods listing;

66
```

In [15]:
```
DATA CARS_DUP;
KEEP TYPE ORIGIN DRIVETRAIN;
SET SASHELP.CARS;
RUN;

PROC PRINT DATA = CARS_DUP(OBS=20);
RUN;
```

Out[15]:

The SAS System

Obs	Type	Origin	DriveTrain
1	SUV	Asia	All
2	Sedan	Asia	Front
3	Sedan	Asia	Front
4	Sedan	Asia	Front
5	Sedan	Asia	Front
6	Sedan	Asia	Front
7	Sports	Asia	Rear
8	Sedan	Europe	Front
9	Sedan	Europe	Front
10	Sedan	Europe	Front
11	Sedan	Europe	All
12	Sedan	Europe	All
13	Sedan	Europe	Front
14	Sedan	Europe	All
15	Sedan	Europe	Front
16	Sedan	Europe	All
17	Sedan	Europe	All
18	Sedan	Europe	All
19	Sedan	Europe	All
20	Sedan	Europe	All

In [16]:
```
PROC SORT DATA = CARS_DUP(OBS=20) OUT = CARS_NDK NODUPKEY;
BY TYPE ORIGIN;
RUN;

PROC PRINT DATA = CARS_NDK;
RUN;
```

Out[16]:

The SAS System

Obs	Type	Origin	DriveTrain
1	SUV	Asia	All
2	Sedan	Asia	Front
3	Sedan	Europe	Front
4	Sports	Asia	Rear

In [17]:
```
PROC SORT DATA = CARS_DUP(OBS=20) OUT = CARS_NQK NOUNIQUEKEY;
BY TYPE ORIGIN;
RUN;

PROC PRINT DATA = CARS_NQK;
RUN;
```

Out[17]:

The SAS System

Obs	Type	Origin	DriveTrain
1	Sedan	Asia	Front
2	Sedan	Asia	Front
3	Sedan	Asia	Front
4	Sedan	Asia	Front
5	Sedan	Asia	Front
6	Sedan	Europe	Front
7	Sedan	Europe	Front
8	Sedan	Europe	Front
9	Sedan	Europe	All
10	Sedan	Europe	All
11	Sedan	Europe	Front
12	Sedan	Europe	All
13	Sedan	Europe	Front
14	Sedan	Europe	All
15	Sedan	Europe	All
16	Sedan	Europe	All
17	Sedan	Europe	All
18	Sedan	Europe	All

In [18]:
```
PROC SORT DATA = CARS_DUP(OBS=20) OUT = CARS_ND NODUP;
BY TYPE ORIGIN;
RUN;

PROC PRINT DATA = CARS_ND;
RUN;
```

Out[18]:

The SAS System

Obs	Type	Origin	DriveTrain
1	SUV	Asia	All
2	Sedan	Asia	Front
3	Sedan	Europe	Front
4	Sedan	Europe	All
5	Sedan	Europe	Front
6	Sedan	Europe	All
7	Sedan	Europe	Front
8	Sedan	Europe	All
9	Sports	Asia	Rear

PROC FREQ

(1)基本形式:

PROC FREQ DATA=;TABLE ;RUN;

对一个变量计算频数，称作ONE-WAY；

对两个变量计算频数，称作TWP-WAY；

对多个变量计算频数，称作交叉表；

(2)TABLE VAR后常接选项:

NOCOL:不打印列百分比

NOROW:不打印行百分比

NOPERCENT:不打印百分比

MISSING:统计缺失值

/OUT:将频次表输出到数据集中

(3)TABLE TYPE:

TABLE COL_A

TABLE COL_A * COL_B

TABLE (COL_A COL_C) * COL_B

TABLE COL_A * COL_B / NOROW NOCOL NOPERCENT

TABLE COL_A * COL_B / MISSING

In [19]:
```
/*      PROC FREQ*/
PROC FREQ DATA = CARS NOPRINT;
TABLE DRIVETRAIN / OUT = CARS_FREQ;
RUN;

PROC PRINT DATA = CARS_FREQ;
RUN;
```

Out[19]:

The SAS System

Obs	DriveTrain	COUNT	PERCENT
1	All	92	21.4953
2	Front	226	52.8037
3	Rear	110	25.7009

In [20]:
```
PROC FREQ DATA = CARS;
TABLE DRIVETRAIN * ORIGIN/NOCOL NOROW NOPERCENT MISSING;
RUN;
```

Out[20]:

The SAS System

The FREQ Procedure

Table of DriveTrain by Origin

DriveTrain \ Origin	Asia	Europe	USA	Total
All	34	36	22	92
Front	99	37	90	226
Rear	25	50	35	110
Total	158	123	147	428

Frequency

In [21]:
```
PROC FREQ DATA = CARS;
TABLE ORIGIN * DRIVETRAIN/NOCOL NOROW NOPERCENT MISSING;
RUN;
```

Out[21]:

The SAS System

The FREQ Procedure

Table of Origin by DriveTrain

Origin \ DriveTrain	All	Front	Rear	Total
Asia	34	99	25	158
Europe	36	37	50	123
USA	22	90	35	147
Total	92	226	110	428

Frequency

In [22]:
```
PROC FREQ DATA = CARS;
TABLE DRIVETRAIN * ORIGIN;
WHERE TYPE = "SUV";
RUN;
```

Out[22]:

The SAS System

The FREQ Procedure

Table of DriveTrain by Origin

DriveTrain	Origin			
	Asia	Europe	USA	Total
All	16 26.67 42.11 64.00	10 16.67 26.32 100.00	12 20.00 31.58 48.00	38 63.33
Front	9 15.00 40.91 36.00	0 0.00 0.00 0.00	13 21.67 59.09 52.00	22 36.67
Total	25 41.67	10 16.67	25 41.67	60 100.00

Frequency
Percent
Row Pct
Col Pct

In [26]:
```
PROC FREQ DATA = CARS(OBS=20);
TABLE TYPE * DRIVETRAIN * ORIGIN;
RUN;
```

Out[26]:

The SAS System

The FREQ Procedure

Table 1 of DriveTrain by Origin

Controlling for Type=SUV

DriveTrain	Origin		
	Asia	Europe	Total
All	1 100.00 100.00 100.00	0 0.00 0.00 .	1 100.00
Front	0 0.00 . 0.00	0 0.00 . .	0 0.00
Rear	0 0.00 . 0.00	0 0.00 . .	0 0.00
Total	1 100.00	0 0.00	1 100.00

Frequency
Percent
Row Pct
Col Pct

Table 2 of DriveTrain by Origin

Controlling for Type=Sedan

DriveTrain	Origin		
	Asia	Europe	Total

All	0 0.00 0.00 0.00	8 44.44 100.00 61.54	8 44.44
Front	5 27.78 50.00 100.00	5 27.78 50.00 38.46	10 55.56
Rear	0 0.00 . 0.00	0 0.00 . 0.00	0 0.00
Total	5 27.78	13 72.22	18 100.00

Frequency
Percent
Row Pct
Col Pct

Table 3 of DriveTrain by Origin

Controlling for Type=Sports

DriveTrain	Origin		
	Asia	**Europe**	**Total**
All	0 0.00 . 0.00	0 0.00 . .	0 0.00
Front	0 0.00 . 0.00	0 0.00 . .	0 0.00
Rear	1 100.00 100.00 100.00	0 0.00 0.00 .	1 100.00
Total	1 100.00	0 0.00	1 100.00

Frequency
Percent
Row Pct
Col Pct

In [27]:
```
PROC FREQ DATA = CARS;
TABLE (TYPE DRIVETRAIN) * ORIGIN;
RUN;
```

Out[27]:

The SAS System

The FREQ Procedure

Table of Type by Origin

Type	Origin: Asia	Europe	USA	Total
Hybrid	3	0	0	3
	0.70	0.00	0.00	0.70
	100.00	0.00	0.00	
	1.90	0.00	0.00	
SUV	25	10	25	60
	5.84	2.34	5.84	14.02
	41.67	16.67	41.67	
	15.82	8.13	17.01	
Sedan	94	78	90	262
	21.96	18.22	21.03	61.21
	35.88	29.77	34.35	
	59.49	63.41	61.22	
Sports	17	23	9	49
	3.97	5.37	2.10	11.45
	34.69	46.94	18.37	
	10.76	18.70	6.12	
Truck	8	0	16	24
	1.87	0.00	3.74	5.61
	33.33	0.00	66.67	
	5.06	0.00	10.88	
Wagon	11	12	7	30
	2.57	2.80	1.64	7.01
	36.67	40.00	23.33	
	6.96	9.76	4.76	
Total	158	123	147	428
	36.92	28.74	34.35	100.00

Frequency
Percent
Row Pct
Col Pct

Table of DriveTrain by Origin

DriveTrain	Origin: Asia	Europe	USA	Total
All	34	36	22	92
	7.94	8.41	5.14	21.50
	36.96	39.13	23.91	
	21.52	29.27	14.97	
Front	99	37	90	226
	23.13	8.64	21.03	52.80
	43.81	16.37	39.82	
	62.66	30.08	61.22	
Rear	25	50	35	110
	5.84	11.68	8.18	25.70
	22.73	45.45	31.82	
	15.82	40.65	23.81	
Total	158	123	147	428
	36.92	28.74	34.35	100.00

Frequency
Percent
Row Pct
Col Pct

PROC UNIVARIATE

(1)基本形式:

PROC UNIVARIATE DATA = ;VAR ;RUN;

单变量过程，可以求单个变量的大部分统计指标

In [29]:
```
/* PROC UNIVARIATE */
PROC UNIVARIATE DATA = CARS;
VAR MSRP;
RUN;
```

Out[29]:

The SAS System

The UNIVARIATE Procedure
Variable: MSRP

Moments

N	428	Sum Weights	428
Mean	32774.8551	Sum Observations	14027638
Std Deviation	19431.7167	Variance	377591613
Skewness	2.79809927	Kurtosis	13.8792055
Uncorrected SS	6.20985E11	Corrected SS	1.61232E11
Coeff Variation	59.2884899	Std Error Mean	939.267478

Basic Statistical Measures

Location		Variability	
Mean	32774.86	Std Deviation	19432
Median	27635.00	Variance	377591613
Mode	13270.00	Range	182185
		Interquartile Range	18886

Note: The mode displayed is the smallest of 18 modes with a count of 2.

Tests for Location: Mu0=0

Test	Statistic		p Value	
Student's t	t	34.89406	Pr > |t|	<.0001
Sign	M	214	Pr >= |M|	<.0001
Signed Rank	S	45903	Pr >= |S|	<.0001

Quantiles (Definition 5)

Level	Quantile
100% Max	192465.0
99%	94820.0
95%	73195.0
90%	52795.0
75% Q3	39215.0

50% Median	27635.0
25% Q1	20329.5
10%	15460.0
5%	13670.0
1%	11155.0
0% Min	10280.0

Extreme Observations

Lowest		Highest	
Value	Obs	Value	Obs
10280	207	94820	262
10539	169	121770	271
10760	383	126670	272
10995	346	128420	263
11155	208	192465	335

In [32]:
```
PROC UNIVARIATE DATA = CARS;
VAR MAKE;
RUN;
```

Out[32]:

```
377  ods listing close;ods html5 (id=saspy_internal) file=stdout options(bitmap_mode='inline') devi
ce=svg style=HTMLBlue; ods
377! graphics on / outputfmt=png;
NOTE: Writing HTML5(SASPY_INTERNAL) Body file: STDOUT
378
379  PROC UNIVARIATE DATA = CARS;
ERROR: Variable Make in list does not match type prescribed for this list.
380  VAR MAKE;
381  RUN;
NOTE: The SAS System stopped processing this step because of errors.
NOTE: PROCEDURE UNIVARIATE used (Total process time):
      real time           0.00 seconds
      cpu time            0.00 seconds

382
383  ods html5 (id=saspy_internal) close;ods listing;

384
```

In [41]:
```
PROC UNIVARIATE DATA = CARS(OBS=100);
VAR MSRP;
CLASS MAKE;
RUN;
```

Out[41]:

The SAS System

The UNIVARIATE Procedure
Variable: MSRP
Make = Acura

Moments

N	7	Sum Weights	7
Mean	42938.5714	Sum Observations	300570
Std Deviation	22189.0077	Variance	492352064
Skewness	1.93342185	Kurtosis	4.2576976
Uncorrected SS	1.58602E10	Corrected SS	2954112386

Coeff Variation	51.6761667	Std Error Mean	8386.65662

Basic Statistical Measures

Location		Variability	
Mean	42938.57	Std Deviation	22189
Median	36945.00	Variance	492352064
Mode	.	Range	65945
		Interquartile Range	19110

Tests for Location: Mu0=0

Test	Statistic		p Value	
Student's t	t	5.119868	Pr > |t|	0.0022
Sign	M	3.5	Pr >= |M|	0.0156
Signed Rank	S	14	Pr >= |S|	0.0156

Quantiles (Definition 5)

Level	Quantile
100% Max	89765
99%	89765
95%	89765
90%	89765
75% Q3	46100
50% Median	36945
25% Q1	26990
10%	23820
5%	23820
1%	23820
0% Min	23820

Extreme Observations

Lowest		Highest	
Value	Obs	Value	Obs
23820	2	33195	4
26990	3	36945	1
33195	4	43755	5
36945	1	46100	6
43755	5	89765	7

The UNIVARIATE Procedure
Variable: MSRP
Make = Audi

Moments			
N	19	Sum Weights	19
Mean	43307.8947	Sum Observations	822850
Std Deviation	13533.6632	Variance	183160040
Skewness	1.96670266	Kurtosis	4.50640655
Uncorrected SS	3.89328E10	Corrected SS	3296880716
Coeff Variation	31.2498755	Std Error Mean	3104.83528

Basic Statistical Measures			
Location		Variability	
Mean	43307.89	Std Deviation	13534
Median	40590.00	Variance	183160040
Mode	35940.00	Range	58660
		Interquartile Range	12100

Tests for Location: Mu0=0				
Test	Statistic		p Value	
Student's t	t	13.94853	Pr > |t|	<.0001
Sign	M	9.5	Pr >= |M|	<.0001
Signed Rank	S	95	Pr >= |S|	<.0001

Quantiles (Definition 5)	
Level	Quantile
100% Max	84600
99%	84600
95%	84600
90%	69190
75% Q3	48040
50% Median	40590
25% Q1	35940
10%	31840
5%	25940
1%	25940
0% Min	25940

Extreme Observations			
Lowest		Highest	

Value	Obs	Value	Obs
25940	8	48040	20
31840	10	49090	26
33430	11	49690	18
34480	12	69190	19
35940	22	84600	21

The UNIVARIATE Procedure
Variable: MSRP
Make = BMW

Moments			
N	20	Sum Weights	20
Mean	43285.25	Sum Observations	865705
Std Deviation	12459.7565	Variance	155245533
Skewness	1.15467662	Kurtosis	0.77055519
Uncorrected SS	4.04219E10	Corrected SS	2949665124
Coeff Variation	28.7852248	Std Error Mean	2786.08626

Basic Statistical Measures			
Location		Variability	
Mean	43285.25	Std Deviation	12460
Median	38995.00	Variance	155245533
Mode	.	Range	44700
		Interquartile Range	15500

Tests for Location: Mu0=0				
Test	Statistic		p Value	
Student's t	t	15.53622	Pr > |t|	<.0001
Sign	M	10	Pr >= |M|	<.0001
Signed Rank	S	105	Pr >= |S|	<.0001

Quantiles (Definition 5)	
Level	Quantile
100% Max	73195
99%	73195
95%	71195
90%	62895
75% Q3	50195
50% Median	38995

25% Q1	34695
10%	30520
5%	29370
1%	28495
0% Min	28495

Extreme Observations

Lowest		Highest	
Value	Obs	Value	Obs
28495	29	52195	28
30245	32	54995	39
30795	30	56595	43
32845	46	69195	40
33895	44	73195	41

The UNIVARIATE Procedure
Variable: MSRP
Make = Buick

Moments

N	9	Sum Weights	9
Mean	30537.7778	Sum Observations	274840
Std Deviation	6371.65604	Variance	40598000.7
Skewness	0.42212948	Kurtosis	-1.1975533
Uncorrected SS	8717786850	Corrected SS	324784006
Coeff Variation	20.864832	Std Error Mean	2123.88535

Basic Statistical Measures

Location		Variability	
Mean	30537.78	Std Deviation	6372
Median	28345.00	Variance	40598001
Mode	.	Range	18540
		Interquartile Range	9075

Tests for Location: Mu0=0

Test	Statistic		p Value	
Student's t	t	14.37826	Pr > |t|	<.0001
Sign	M	4.5	Pr >= |M|	0.0039
Signed Rank	S	22.5	Pr >= |S|	0.0039

Quantiles (Definition 5)	
Level	Quantile
100% Max	40720
99%	40720
95%	40720
90%	40720
75% Q3	35545
50% Median	28345
25% Q1	26470
10%	22180
5%	22180
1%	22180
0% Min	22180

Extreme Observations			
Lowest		Highest	
Value	Obs	Value	Obs
22180	49	28345	52
24895	51	32245	53
26470	50	35545	54
26545	48	37895	47
28345	52	40720	55

The UNIVARIATE Procedure
Variable: MSRP
Make = Cadillac

Moments			
N	8	Sum Weights	8
Mean	50474.375	Sum Observations	403795
Std Deviation	12552.2549	Variance	157559103
Skewness	0.89503837	Kurtosis	3.21609142
Uncorrected SS	2.14842E10	Corrected SS	1102913722
Coeff Variation	24.8685693	Std Error Mean	4437.89228

Basic Statistical Measures			
Location		Variability	
Mean	50474.38	Std Deviation	12552
Median	49275.00	Variance	157559103
Mode	.	Range	45365

		Interquartile Range	6665

Tests for Location: Mu0=0				
Test	Statistic		p Value	
Student's t	t	11.3735	Pr > |t|	<.0001
Sign	M	4	Pr >= |M|	0.0078
Signed Rank	S	18	Pr >= |S|	0.0078

Quantiles (Definition 5)	
Level	Quantile
100% Max	76200
99%	76200
95%	76200
90%	76200
75% Q3	52885
50% Median	49275
25% Q1	46220
10%	30835
5%	30835
1%	30835
0% Min	30835

Extreme Observations			
Lowest		Highest	
Value	Obs	Value	Obs
30835	58	47955	61
45445	59	50595	60
46995	57	52795	56
47955	61	52975	63
50595	60	76200	62

The UNIVARIATE Procedure
Variable: MSRP
Make = Chevrolet

Moments			
N	27	Sum Weights	27
Mean	26587.037	Sum Observations	717850
Std Deviation	10887.9446	Variance	118547337
Skewness	0.75702534	Kurtosis	-0.4136664

Uncorrected SS	2.21677E10	**Corrected SS**	3082230763
Coeff Variation	40.9520796	**Std Error Mean**	2095.38591

Basic Statistical Measures

Location		Variability	
Mean	26587.04	**Std Deviation**	10888
Median	23495.00	**Variance**	118547337
Mode	.	**Range**	39845
		Interquartile Range	17105

Tests for Location: Mu0=0

Test	Statistic		p Value	
Student's t	**t**	12.68837	**Pr > |t|**	<.0001
Sign	**M**	13.5	**Pr >= |M|**	<.0001
Signed Rank	**S**	189	**Pr >= |S|**	<.0001

Quantiles (Definition 5)

Level	Quantile
100% Max	51535
99%	51535
95%	44535
90%	42735
75% Q3	36100
50% Median	23495
25% Q1	18995
10%	14610
5%	12585
1%	11690
0% Min	11690

Extreme Observations

Lowest		Highest	
Value	**Obs**	**Value**	**Obs**
11690	68	41465	65
12585	69	41995	89
14610	70	42735	64
14810	71	44535	83
16385	72	51535	84

The UNIVARIATE Procedure
Variable: MSRP
Make = Chrysler

Moments			
N	10	Sum Weights	10
Mean	24623.5	Sum Observations	246235
Std Deviation	4706.45423	Variance	22150711.4
Skewness	0.41043306	Kurtosis	-0.1470545
Uncorrected SS	6262523925	Corrected SS	199356403
Coeff Variation	19.1136688	Std Error Mean	1488.31151

Basic Statistical Measures			
Location		Variability	
Mean	24623.50	Std Deviation	4706
Median	24672.50	Variance	22150711
Mode	.	Range	15310
		Interquartile Range	5020

Tests for Location: Mu0=0				
Test	Statistic		p Value	
Student's t	t	16.54459	Pr > |t|	<.0001
Sign	M	5	Pr >= |M|	0.0020
Signed Rank	S	27.5	Pr >= |S|	0.0020

Quantiles (Definition 5)	
Level	Quantile
100% Max	33295.0
99%	33295.0
95%	33295.0
90%	31580.0
75% Q3	26860.0
50% Median	24672.5
25% Q1	21840.0
10%	18537.5
5%	17985.0
1%	17985.0
0% Min	17985.0

Extreme Observations	

	Lowest		Highest	
	Value	Obs	Value	Obs
	17985	91	25215	99
	19090	93	25955	98
	21840	94	26860	97
	22000	92	29865	95
	24130	96	33295	100

PROC TRANSPOSE

(1)基本形式:

PROC TRANSPOSE DATA = OUT = ;VAR ;RUN;

实现对数据集的转置，即把观测变为变量，变量变为观测；

(2)常用选项:

PREFIX:规定转置后的变量名前缀；

SUFFIX:规定转置后的变量名后缀；

OUT:规定输出数据集；

ID:使用其后规定的变量值作为输出数据集中被转置的变量名；

VAR:规定需要转置的变量名；

BY:规定分组的变量名；

In [35]:
```
/* PROC TRANSPOSE */
DATA CARS_MAMS;
KEEP MAKE TYPE MSRP;
SET SASHELP.CARS;
RUN;

PROC SORT DATA = CARS_MAMS OUT = CARS_MAMSD NODUPKEY;
BY MAKE TYPE;
RUN;

PROC PRINT DATA = CARS_MAMSD;
RUN;
```

Out[35]:

The SAS System

Obs	Make	Type	MSRP
1	Acura	SUV	$36,945
2	Acura	Sedan	$23,820
3	Acura	Sports	$89,765
4	Audi	Sedan	$25,940
5	Audi	Sports	$84,600
6	Audi	Wagon	$40,840
7	BMW	SUV	$37,000
8	BMW	Sedan	$28,495
9	BMW	Sports	$48,195
10	BMW	Wagon	$32,845
11	Buick	SUV	$37,895

12	Buick	Sedan	$22,180
13	Cadillac	SUV	$52,795
14	Cadillac	Sedan	$30,835
15	Cadillac	Sports	$76,200
16	Cadillac	Truck	$52,975
17	Chevrolet	SUV	$42,735
18	Chevrolet	Sedan	$11,690
19	Chevrolet	Sports	$44,535
20	Chevrolet	Truck	$36,100
21	Chevrolet	Wagon	$22,225
22	Chrysler	Sedan	$17,985
23	Chrysler	Sports	$34,495
24	Chrysler	Wagon	$31,230
25	Dodge	SUV	$32,235
26	Dodge	Sedan	$13,670
27	Dodge	Sports	$81,795
28	Dodge	Truck	$17,630
29	Ford	SUV	$41,475
30	Ford	Sedan	$13,270
31	Ford	Sports	$18,345
32	Ford	Truck	$22,010
33	Ford	Wagon	$17,475
34	GMC	SUV	$31,890
35	GMC	Sedan	$25,640
36	GMC	Truck	$16,530
37	Honda	Hybrid	$20,140
38	Honda	SUV	$27,560
39	Honda	Sedan	$13,270
40	Honda	Sports	$33,260
41	Hummer	SUV	$49,995
42	Hyundai	SUV	$21,589
43	Hyundai	Sedan	$10,539
44	Hyundai	Sports	$18,739
45	Infiniti	Sedan	$28,495
46	Infiniti	Wagon	$34,895
47	Isuzu	SUV	$31,849
48	Jaguar	Sedan	$29,995
49	Jaguar	Sports	$69,995
50	Jeep	SUV	$27,905
51	Kia	SUV	$19,635
52	Kia	Sedan	$16,040
53	Kia	Wagon	$11,905

54	Land Rover	SUV	$72,250
55	Lexus	SUV	$45,700
56	Lexus	Sedan	$32,350
57	Lexus	Sports	$63,200
58	Lexus	Wagon	$32,455
59	Lincoln	SUV	$52,775
60	Lincoln	Sedan	$32,495
61	MINI	Sedan	$16,999
62	Mazda	SUV	$21,087
63	Mazda	Sedan	$15,500
64	Mazda	Sports	$22,388
65	Mazda	Truck	$14,840
66	Mercedes-Benz	SUV	$76,870
67	Mercedes-Benz	Sedan	$26,060
68	Mercedes-Benz	Sports	$90,520
69	Mercedes-Benz	Wagon	$33,780
70	Mercury	SUV	$29,995
71	Mercury	Sedan	$21,595
72	Mercury	Wagon	$22,595
73	Mitsubishi	SUV	$30,492
74	Mitsubishi	Sedan	$14,622
75	Mitsubishi	Sports	$25,092
76	Mitsubishi	Wagon	$17,495
77	Nissan	SUV	$33,840
78	Nissan	Sedan	$12,740
79	Nissan	Sports	$26,910
80	Nissan	Truck	$19,479
81	Nissan	Wagon	$28,739
82	Oldsmobile	Sedan	$18,825
83	Pontiac	SUV	$21,595
84	Pontiac	Sedan	$15,495
85	Pontiac	Sports	$33,500
86	Pontiac	Wagon	$17,045
87	Porsche	SUV	$56,665
88	Porsche	Sports	$79,165
89	Saab	Sedan	$30,860
90	Saab	Wagon	$40,845
91	Saturn	SUV	$20,585
92	Saturn	Sedan	$10,995
93	Saturn	Wagon	$23,560
94	Scion	Sedan	$12,965

95	Scion	Wagon	$14,165
96	Subaru	Sedan	$19,945
97	Subaru	Sports	$25,045
98	Subaru	Truck	$24,520
99	Subaru	Wagon	$21,445
100	Suzuki	SUV	$23,699
101	Suzuki	Sedan	$12,884
102	Suzuki	Wagon	$16,497
103	Toyota	Hybrid	$20,510
104	Toyota	SUV	$35,695
105	Toyota	Sedan	$14,085
106	Toyota	Sports	$22,570
107	Toyota	Truck	$12,800
108	Toyota	Wagon	$16,695
109	Volkswagen	SUV	$35,515
110	Volkswagen	Sedan	$18,715
111	Volkswagen	Wagon	$19,005
112	Volvo	SUV	$41,250
113	Volvo	Sedan	$25,135
114	Volvo	Wagon	$26,135

In [39]:
```
PROC TRANSPOSE DATA = CARS_MAMSD;
VAR MSRP;
BY MAKE;
RUN;

/* DATAX */
```

Out[39]:

```
457  ods listing close;ods html5 (id=saspy_internal) file=stdout options(bitmap_mode='inline') devi
ce=svg style=HTMLBlue; ods
457! graphics on / outputfmt=png;
NOTE: Writing HTML5(SASPY_INTERNAL) Body file: STDOUT
458
459  PROC TRANSPOSE DATA = CARS_MAMSD;
460  VAR MSRP;
461  BY MAKE;
462  RUN;
NOTE: There were 114 observations read from the data set WORK.CARS_MAMSD.
NOTE: The data set WORK.DATA4 has 38 observations and 8 variables.
NOTE: Compressing data set WORK.DATA4 increased size by 100.00 percent.
      Compressed is 2 pages; un-compressed would require 1 pages.
NOTE: PROCEDURE TRANSPOSE used (Total process time):
      real time           0.00 seconds
      cpu time            0.00 seconds

463
464  ods html5 (id=saspy_internal) close;ods listing;

465
```

In [40]:
```
PROC TRANSPOSE DATA = CARS_MAMSD OUT = CARS_M PREFIX = ID_;
VAR MSRP;
BY MAKE;
ID TYPE;
RUN;

PROC PRINT DATA = CARS_M;
RUN;
```

Out[40]:

Out[40]:

The SAS System

Obs	Make	_NAME_	ID_SUV	ID_Sedan	ID_Sports	ID_Wagon	ID_Truck	ID_Hybrid
1	Acura	MSRP	$36,945	$23,820	$89,765	.	.	.
2	Audi	MSRP	.	$25,940	$84,600	$40,840	.	.
3	BMW	MSRP	$37,000	$28,495	$48,195	$32,845	.	.
4	Buick	MSRP	$37,895	$22,180	.	.	.	.
5	Cadillac	MSRP	$52,795	$30,835	$76,200	.	$52,975	.
6	Chevrolet	MSRP	$42,735	$11,690	$44,535	$22,225	$36,100	.
7	Chrysler	MSRP	.	$17,985	$34,495	$31,230	.	.
8	Dodge	MSRP	$32,235	$13,670	$81,795	.	$17,630	.
9	Ford	MSRP	$41,475	$13,270	$18,345	$17,475	$22,010	.
10	GMC	MSRP	$31,890	$25,640	.	.	$16,530	.
11	Honda	MSRP	$27,560	$13,270	$33,260	.	.	$20,140
12	Hummer	MSRP	$49,995	.	.	.	.	.
13	Hyundai	MSRP	$21,589	$10,539	$18,739	.	.	.
14	Infiniti	MSRP	.	$28,495	.	$34,895	.	.
15	Isuzu	MSRP	$31,849	.	.	.	.	.
16	Jaguar	MSRP	.	$29,995	$69,995	.	.	.
17	Jeep	MSRP	$27,905	.	.	.	.	.
18	Kia	MSRP	$19,635	$16,040	.	$11,905	.	.
19	Land Rover	MSRP	$72,250	.	.	.	.	.
20	Lexus	MSRP	$45,700	$32,350	$63,200	$32,455	.	.
21	Lincoln	MSRP	$52,775	$32,495	.	.	.	.
22	MINI	MSRP	.	$16,999	.	.	.	.
23	Mazda	MSRP	$21,087	$15,500	$22,388	.	$14,840	.
24	Mercedes-Benz	MSRP	$76,870	$26,060	$90,520	$33,780	.	.
25	Mercury	MSRP	$29,995	$21,595	.	$22,595	.	.
26	Mitsubishi	MSRP	$30,492	$14,622	$25,092	$17,495	.	.
27	Nissan	MSRP	$33,840	$12,740	$26,910	$28,739	$19,479	.
28	Oldsmobile	MSRP	.	$18,825	.	.	.	.
29	Pontiac	MSRP	$21,595	$15,495	$33,500	$17,045	.	.
30	Porsche	MSRP	$56,665	.	$79,165	.	.	.
31	Saab	MSRP	.	$30,860	.	$40,845	.	.
32	Saturn	MSRP	$20,585	$10,995	.	$23,560	.	.
33	Scion	MSRP	.	$12,965	.	$14,165	.	.
34	Subaru	MSRP	.	$19,945	$25,045	$21,445	$24,520	.
35	Suzuki	MSRP	$23,699	$12,884	.	$16,497	.	.
36	Toyota	MSRP	$35,695	$14,085	$22,570	$16,695	$12,800	$20,510
37	Volkswagen	MSRP	$35,515	$18,715	.	$19,005	.	.
38	Volvo	MSRP	$41,250	$25,135	.	$26,135	.	.

PROC SURVEYSELECT

(1)基本形式:

PROC SURVEYSELECT DATA = METHOD = N = ;RUN;

PROC SURVEYSELECT DATA = METHOD = SAMPRATE = ;RUN;

实现对数据的随机抽样；

(2)常用选项:

METHOD：随机抽样的方法

SRS（Simple Random Sampling，不放回简单随机抽样）；

URS（Unrestricted Random Sampling，放回简单随机抽样）；

SYS（Systematic Sampling，系统抽样）；

SEED：随机种子数，随机数产生器；

非负整数，若为0则以当前时间作为当前随机中子数，则可实现每次抽取的样本不同；

若取大于0的整数，则下次抽样时若输入相同值即可得到相同的样本；

In [43]:
```
/* PROC SURVEYSELECT */
PROC SURVEYSELECT
    DATA = CARS_MAMS METHOD = SRS N = 3
    OUT = CARS_SRS_N3;
RUN;

PROC PRINT DATA = CARS_SRS_N3;
RUN;
```

Out[43]:

The SAS System

The SURVEYSELECT Procedure

Selection Method	Simple Random Sampling

Input Data Set	CARS_MAMS
Random Number Seed	286169112
Sample Size	3
Selection Probability	0.007009
Sampling Weight	142.66667
Output Data Set	CARS_SRS_N3

The SAS System

Obs	Make	Type	MSRP
1	Honda	SUV	$19,860
2	Kia	Sedan	$12,360
3	Subaru	Wagon	$21,445

In [46]:
```
PROC SURVEYSELECT
    DATA = CARS_MAMS METHOD = SRS SAMPRATE = 0.01
    OUT = CARS_SRS_P1;
RUN;

PROC PRINT DATA = CARS_SRS_P1;
RUN;
```

Out[46]:

The SAS System

The SURVEYSELECT Procedure

Selection Method	Simple Random Sampling

Input Data Set	CARS_MAMS
Random Number Seed	325312536
Sampling Rate	0.01
Sample Size	5
Selection Probability	0.011682
Sampling Weight	85.6
Output Data Set	CARS_SRS_P1

The SAS System

Obs	Make	Type	MSRP
1	Dodge	Sedan	$21,795
2	Mercedes-Benz	Sedan	$33,480
3	Mercury	SUV	$29,995
4	Mitsubishi	SUV	$33,112
5	Subaru	Sedan	$25,645

In [45]:
```
PROC SURVEYSELECT
    DATA = CARS_MAMS METHOD = SRS SAMPRATE = 0.01
    OUT = CARS_SRS_N1;
STRATA MAKE;
RUN;

PROC PRINT DATA = CARS_SRS_N1;
RUN;
```

Out[45]:

The SAS System

The SURVEYSELECT Procedure

Selection Method	Simple Random Sampling
Strata Variable	Make

Input Data Set	CARS_MAMS
Random Number Seed	312152788
Stratum Sampling Rate	0.1
Number of Strata	38

Total Sample Size	62
Output Data Set	CARS_SRS_N1

The SAS System

Obs	Make	Type	MSRP	SelectionProb	SamplingWeight
1	Acura	Sedan	$43,755	0.14286	7.0000
2	Audi	Sedan	$34,480	0.10526	9.5000
3	Audi	Sports	$84,600	0.10526	9.5000
4	BMW	Sedan	$69,195	0.10000	10.0000
5	BMW	Sports	$33,895	0.10000	10.0000
6	Buick	Sedan	$28,345	0.11111	9.0000
7	Cadillac	Truck	$52,975	0.12500	8.0000
8	Chevrolet	Sedan	$14,610	0.11111	9.0000
9	Chevrolet	Sedan	$20,370	0.11111	9.0000
10	Chevrolet	Sedan	$25,000	0.11111	9.0000
11	Chrysler	Sports	$34,495	0.13333	7.5000
12	Chrysler	Wagon	$31,230	0.13333	7.5000
13	Dodge	Sedan	$24,885	0.15385	6.5000
14	Dodge	Truck	$20,215	0.15385	6.5000
15	Ford	Sedan	$19,135	0.13043	7.6667
16	Ford	Sedan	$30,315	0.13043	7.6667
17	Ford	Sports	$29,380	0.13043	7.6667
18	GMC	Truck	$29,322	0.12500	8.0000
19	Honda	SUV	$19,860	0.11765	8.5000
20	Honda	SUV	$18,690	0.11765	8.5000
21	Hummer	SUV	$49,995	1.00000	1.0000
22	Hyundai	Sedan	$15,389	0.16667	6.0000
23	Hyundai	Sedan	$20,339	0.16667	6.0000
24	Infiniti	Sedan	$29,795	0.12500	8.0000
25	Isuzu	SUV	$31,849	0.50000	2.0000
26	Jaguar	Sedan	$74,995	0.16667	6.0000
27	Jaguar	Sports	$69,995	0.16667	6.0000
28	Jeep	SUV	$20,130	0.33333	3.0000
29	Kia	Sedan	$26,000	0.18182	5.5000
30	Kia	Wagon	$11,905	0.18182	5.5000
31	Land Rover	SUV	$72,250	0.33333	3.0000
32	Lexus	SUV	$64,800	0.18182	5.5000
33	Lexus	SUV	$39,195	0.18182	5.5000
34	Lincoln	Sedan	$44,925	0.11111	9.0000
35	MINI	Sedan	$16,999	0.50000	2.0000

36	Mazda	Sports	$25,193	0.18182	5.5000
37	Mazda	Sports	$25,700	0.18182	5.5000
38	Mercedes-Benz	Sedan	$37,630	0.11538	8.6667
39	Mercedes-Benz	Sedan	$52,120	0.11538	8.6667
40	Mercedes-Benz	Wagon	$33,780	0.11538	8.6667
41	Mercury	Sedan	$21,595	0.11111	9.0000
42	Mitsubishi	Sedan	$25,700	0.15385	6.5000
43	Mitsubishi	Sports	$25,092	0.15385	6.5000
44	Nissan	SUV	$27,339	0.11765	8.5000
45	Nissan	Sedan	$32,780	0.11765	8.5000
46	Oldsmobile	Sedan	$28,790	0.33333	3.0000
47	Pontiac	Sedan	$15,495	0.18182	5.5000
48	Pontiac	Sedan	$24,295	0.18182	5.5000
49	Porsche	Sports	$192,465	0.14286	7.0000
50	Saab	Sedan	$40,670	0.14286	7.0000
51	Saturn	Sedan	$15,825	0.12500	8.0000
52	Scion	Sedan	$12,965	0.50000	2.0000
53	Subaru	Sedan	$29,345	0.18182	5.5000
54	Subaru	Wagon	$21,445	0.18182	5.5000
55	Suzuki	SUV	$17,163	0.12500	8.0000
56	Toyota	SUV	$54,765	0.10714	9.3333
57	Toyota	Sedan	$11,560	0.10714	9.3333
58	Toyota	Sedan	$11,290	0.10714	9.3333
59	Volkswagen	Sedan	$18,715	0.13333	7.5000
60	Volkswagen	Sedan	$21,055	0.13333	7.5000
61	Volvo	Sedan	$31,745	0.16667	6.0000
62	Volvo	Sedan	$34,845	0.16667	6.0000

In [ ]: