Horizontal Data Sorting and Insightful Reporting --- A Useful SAS® Technique

Justin Jia
Trans Union Canada

October 24, 2014
Burlington, Ontario, Canada

Overview

- What is horizontal data sorting?
- Why is it necessary and useful?
- Case Study 1: Student Profiling--- Horizontal Sorting of Numeric Variables
- Case Study 2: Horizontal Sorting of Customer Purchase Data
- Business Application Examples: Retail and Financial
- Conclusion

What is horizontal data sorting?

- **Vertical data sorting:**
  sorts data across rows or observations based on one or more variables.
- **Example: vertical sort by ID, Province and Sales.**

ID	Province	Sales
A01	AB	$900
A01	AB	$500
A01	ON	$700
A01	ON	$300

- **Common and easy to realize in SAS.**
  Proc Sort
  Proc SQL (Order By)

- **Horizontal data sorting**
  sorts data across columns or variables.
- **Less common and NOT easy to do, requiring advanced SAS skills (Do-Loop, Array, Rotate/Transpose etc).**
- **However, it is needed and useful in some cases, and can provide valuable insightful information into data.**

Why needed? A cross-sell example

- **Raw Data:** Two dimensions (product and purchase quantity), no order, very hard to look up or compare.

Client ID	Product 1	Qty	Product 2	Qty	Product 3	Qty	Product 4	Qty
101	Printer	10	Computer	25	Games	6	TV	3
107	TV	17	Software	32	Fax Machine	11		

- **Isn't it better if we report data…**

  In the ascending order of product names?

Client ID	Product 1	Qty	Product 2	Qty	Product 3	Qty	Product 4	Qty
101	Computer	25	Games	6	Printer	10	TV	3
107	Fax Machine	11	Software	32	TV	17		

  Or, in the descending order of purchase quantities?

Client ID	Product 1	Qty	Product 2	Qty	Product 3	Qty	Product 4	Qty
101	Computer	25	Printer	10	Games	6	TV	3
107	Software	32	TV	17	Fax Machine	11		

Case 1: Student Profiling

- **Psychological study:**
  Each student can take any of the 12 available simulated exams: min. 3, max. 12.
- **Raw data of test scores is NOT in any defined order.**

ID	Test_1	Test_2	Test_3	Test_4	Test_5	Test_6	Test_7	Test_8	Test_9	Test_10	Test_11	Test_12
A001	82	99	100	69	34	99	94	42	42	97	.	.
A003	.	45	67	.	45	88	95	.	.	.	63	32
A004	52	.	35	66	.	77	79	.	68	.	58	79
A006	86	82	71	45	78	.	52	.	45	31	.	78

- **It will be better to arrange and report test scores in a sorted order. Then, how to do it?**

Method 1A: Bubble Sort Approach

- **It is a simple but widely used sorting algorithm.**

  Compares each pair of adjacent elements.

  Swaps them if they are in the unwanted order.

  Repeats the process until no swaps are needed.

```
data Sort_A(drop=I J temp);
set P.test;
array S(12) test_1-test_12;
do I=1 to 12;
do J=1 to 12-I;
if S(J) < S(J+1) then do;
*sort in descending order;
Temp = S(J);
S(J)= S(J+1);
S(J+1) = Temp;
end; end; end;
run;
```

- **Horizontally sorted data set SORT_A.**

ID	Test_1	Test_2	Test_3	Test_4	Test_5	Test_6	Test_7	Test_8	Test_9	Test_10	Test_11	Test_12
A001	100	99	99	97	94	82	69	42	42	34	.	.
A003	95	88	67	63	45	45	32	.	.	.	.	.
A004	79	79	77	68	66	58	52	35	.	.	.	.
A006	86	82	78	78	71	52	45	45	31	.	.	.

Method 1B: Rotate &Transpose Approach

- **Use DATA step array to rotate the raw data into vertical form.**
- **PROC SORT to sort data vertically.**
- **PROC TRANSPOSE to transpose the vertically sorted data back to horizontal form.**
- **It produces the same data set as in Method 1A.**

```
proc sort data=P.test;
by ID; run;

data B_rotate(drop=I test_1-test_12);
set P.test;
by ID;
array S(12) test_1-test_12;
if first.ID then do I=1 to 12;
Score=S(I);
if not missing(score) then output;
end;
run;

proc sort data= B_rotate;
by ID descending score; run;

proc transpose data= B_rotate
out=Sort_B(drop=_name_) prefix=test_;
by ID ;
var score;
run;
```

Method 1C: Use LARGEST function

- **LARGEST/SMALLEST functions.**
  New SAS9.0 functions.

  LARGEST(k, V1, V2, V3… Vn)

  SMALLEST(k, V1, V2, V3… Vn)

  It returns the **kth highest (lowest) value** of a list of variables.

- **Much more concise SAS coding.**

```
data Sort_C(drop= T1-T12 J);
set P.test(rename=(Test_1=T1
   Test_2=T2 Test_3=T3 Test_4=T4
   Test_5=T5 Test_6=T6 Test_7=T7
   Test_8=T8 Test_9=T9 Test_10=T10
   Test_11=T11 Test_12=T12)) ;

array Test(12) Test_1 - Test_12;

do J=1 to 12;
Test(J)= largest(J, of T1-T12);
*Test(J)=smallest(13-J,of T1-T12);
end;
run;
```

Method 1D: Use CALL SORTN routine

- **CALL SORTN/SORTC**
  New call routines as of SAS 9.2.

  CALL SORTN(V1, V2, V3… Vn)

  CALL SORTC(V1, V2, V3… Vn).

- **They always sort data in ascending order**
  However, reverse the order of variables can lead to descending sort.
  CALL SORTN(Vn, Vn-1…V3, V2, V1).

```
data Sort_D;
set P.test;
call sortN(of Test_12-Test_1);
run;
```

- **The simplest method among all**
  Most concise in coding and efficient in performance.

Performance Comparison

Method	Real Time
A: Bubble Sort	32.7s
B: Rotate and Transpose	52.6s
C: LARGEST function	14.8s
D: Call SortN routine	8.8s

Test dataset: 17 num vars, 3.5M obs.

Case 2: Horizontal Sort of Cross-Sell Data

- **Raw data: two dimensions, orderless**

Client ID	Product 1	Qty	Product 2	Qty	Product 3	Qty	Product 4	Qty
101	Printer	10	Computer	25	Games	6	TV	3
107	TV	17	Software	32	Fax Machine	11		

- **A better reporting through horizontal sorting**

Client ID	Product 1	Qty	Product 2	Qty	Product 3	Qty	Product 4	Qty
101	Computer	25	Printer	10	Games	6	TV	3
107	Software	32	TV	17	Fax Machine	11		

- **Challenges:**

  --- Product Name: char. Purchase Quantity: num.

  --- Two dimensions are **relevant to each other rather than independent of each other**. We can NOT sort and arrange them based on only one dimension, it will mess up the data.

- **Three workable and generalizable methods**

  2A: Bubble Sort. 2B: Rotate and Transpose. 2C: Call SortN/C.

- **Please refer to SAS Paper 376-2013 for details.**

Business Application: Retail Industry

- **RFM Analysis:** A useful analytic methodology, often used in retail business analytics:
  simple, quick, low cost, easy to interpret results.
  R: purchase recency. F: purchase frequency.
  M: purchase monetary value ($).
- **RFM Analyses of Online Purchases:**
  By Descending Purchase Value.

Client ID	Most Monetary Purchase Category	Purchase Value($)	2nd Most Monetary Purchase Category	Purchase Value($)	3rd Most Monetary Purchase Category	Purchase Value($)
101	Patio & Garden	$2,752	Auto Accessories	$2,483	Household Essentials	$2,387
126	Computers	$2,524	Sports	$2,404	Jewelry	$1,938
203	Household Essentials	$2,774	Clothes	$2,377	Exercise & Fitness	$1,550

- **Horizontal sorting provides competitive insights into customer purchase behaviours/preferences.**

Business Application: Financial Industry

- **Product Purchase Sequence Analyses:**
  Sort by client ID and opened date.

Client_ID	Product	Opened_Date	Product	Opened_Date	Product	Opened_Date	Product	Opened_Date
111	SAVINGS	2010-11-29	VISA	2012-08-08	MTG	2012-09-18		
112	**CHEQUING**	2012-06-11	TFSA	2012-10-24	LOAN	2012-10-25	VISA	2012-10-25
113	LOAN	2011-02-28	RESP	2011-02-28	CHEQUING	2011-09-14	PLC	2012-03-27
114	TFSA	2010-02-04	PLC	2010-04-30	RESP	2010-12-12	RRSP	2011-12-12
115	**CHEQUING**	2011-05-26	TFSA	2011-05-26	SAVINGS	2011-06-01		

- **A simple frequency distribution gives us the ranking and preference of purchased products.**

Ranking	1st Purchased Product	Clients(#)	Clients(%)	2nd Purchased Product	Clients(#)	Clients(%)	3rd Purchased Product	Clients(#)	Clients(%)
1	Chequing	580,597	22.7%	Savings	318,171	12.8%	Credit Card	181,995	7.1%
2	Credit Card	304,540	13.3%	Credit Card	291,198	11.6%	Savings	173,087	6.5%
3	Savings	304,121	13.2%	Credit Insurance	280,253	11.3%	TFSA	160,478	6.3%
4	TFSA	281,670	12.2%	TFSA	249,403	10.5%	Credit Insurance	136,779	5.4%
5	Mortgage	239,208	10.7%	Chequing	158,380	8.3%	Chequing	87,606	4.1%

Next Best Offer Analysis

Customers With One Current Product

Current Product	Next Purchased Product (Top 5)					
Chequing	Product	Savings	Credit Card	Credit Insurance	TFSA	Loan
	% of Clients	32.8%	20.7%	11.2%	10.9%	7.5%
Credit Card	Product	Chequing	Savings	TFSA	Mortgage	Line of Credit
	% of Clients	28.9%	20.1%	15.7%	13.8%	6.3%
Savings	Product	Chequing	TFSA	Mortgage	GIC	Credit Card
	% of Clients	40.7%	19.4%	13.5%	9.8%	5.2%

Customers With Two Current Products

Current Products	Next Purchased Product (Top 3)			
Chequing + Savings	Product	TFSA	Credit Card	Credit Insurance
	% of Clients	29.1%	15.9%	11.3%
Chequing + Credit Card	Product	Savings	TFSA	Mortgage
	% of Clients	31.5%	28.7%	15.9%
Savings + Credit Card	Product	Chequing	TFSA	Line of Credit
	% of Clients	42.7%	21.8%	10.5%

When to provide the next offer?

- **Purchase Time Span Analysis**

Current Product	Next Product	Time Span To Purchase Next Products (days)			
		N	Min	Max	Mean
Chequing	**Overall**	2,135,673	0	636	**183.8**
	Savings	704,772	1	507	118.2
	Credit Card	491,205	2	325	**93.5**
	TFSA	234,924	26	629	254.6
Credit Card		N	Min	Max	Mean
	Overall	2,236,539	0	568	**257.6**
	Chequing	626,231	0	512	**192.1**
	Savings	469,673	1	533	216.0
	TFSA	351,137	68	554	235.8

- **We can therefore probe and understand customers' banking preferences and business patterns, and gain insights in designing competitive marketing strategies.**

Conclusion

- Horizontal data sorting is a very useful SAS® technique in advanced data analysis and reporting.
- Utilization of this technique can significantly enhance the format and layout of data analysis and presentation.
- More important, it helps to leverage data and provide competitive insights into customer and business data.
- It can have important applications in a wide variety of business analytics and business intelligence fields.

Acknowledgement

- **Financial sponsorship from CIBC and SAS® Canada.**

- **Thank you so much!**