Paper 352-2012

Wallet and Share of Wallet Estimation: A Flexible Methodology

Shira Guil Witelson, New York, NY

ABSTRACT

In this paper, we will demonstrate how Wallet and Share of Wallet estimation, widely used in the credit card industry for marketing and sales strategy, can be applied in a range of industries. The methodology presented consists of a series of steps, beginning with primary research, and progressing to the use of SAS®-enabled unsupervised learning method, predictive modeling and statistical test techniques. The result is a flexible means for estimating the most realistically attainable Wallet of a customer and thereby calculating the customer's Share of Wallet.

Definitions:

Internal Spend: The dollar amount the customer spends on the company's products and services.

External Spend: The dollar amount the customer spends on other providers' products and services.

Wallet: The Total Spend (Internal and External) made by the customer with the company and other providers, so long as the company provides one of the services purchased.

Formulas:

Wallet = Internal Spend + External Spend

Share of Wallet = Internal Spend / Wallet

INTRODUCTION

Wallet and Share of Wallet estimation can have a dramatic impact on marketing and sales strategies. The credit card industry routinely uses External Spend data, on which Wallet estimation relies, as these data are readily available to this industry through the credit bureaus. However, Wallet and Share of Wallet estimation are rarely employed in other industries, mainly because External Spend data are unknown.

This paper draws on some of the ideas explored in the study "Wallet Estimation Models" by IBM's Predictive Modeling Group. One is that customers with certain attributes can be predicted to spend as much as top spenders (high Wallet customers) who share the same attributes. Building on this premise, it is possible to estimate the most realistically attainable Wallet for customers sharing similar attributes.

The first portion of this paper describes a five-stage methodology for Wallet estimation, beginning with primary research to collect a sample of Wallet data. Following data preparation, the methodology proceeds to the more advanced Analysis stages, in which we offer an applicative example using the following statistical techniques: Clustering with Proc Fastclus; segment comparison and evaluation with Proc Glm; and Wallet assignment using Proc Univariate.

WALLET ESTIMATION METHODOLOGY

The Wallet estimation process consists of five stages:

1. List Creation
2. Primary Research (survey)
3. Segment Creation
4. Segment Evaluation
5. Wallet Assignment

1. LIST CREATION

Create a list of potential demographic and in-house variables that could affect the customer Wallet. Relevant demographic variables would include Age, Children, Income, Gender and Education. In-house variables might include Tenure and Product Holding, for example.

2. PRIMARY RESEARCH

Conduct a survey among a sample of customers to collect each customer's External Spend. The External Spend is added to the Internal Spend in order to determine each customer's Wallet.

The survey also presents a good opportunity to collect demographic data. In company databases, these data (typically collected by a third party) tend to be unreliable since they are largely estimated or missing altogether. While it is not necessary to collect demographic data with the survey, doing so can yield a clean sample for a range of analytic purposes and is therefore highly recommended.

3. SEGMENT CREATION

The creation of segments consists of two phases, clustering and slicing.

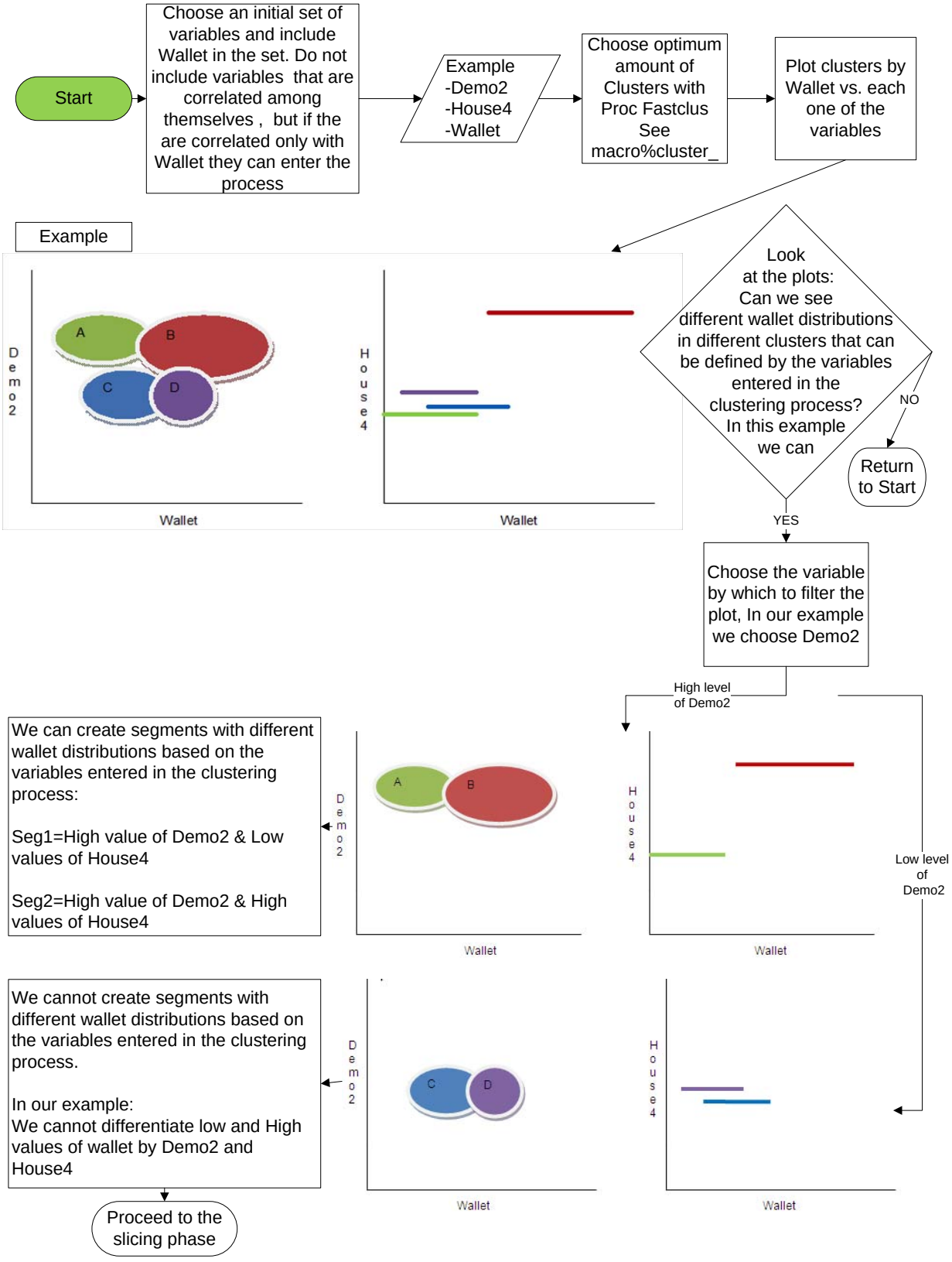
i. **Clustering:** The clustering phase involves creating segments, each of which displays a different Wallet distribution and can be differentiated by a specific combination of variables.

   The primary purpose of clustering is to consolidate similar observations. Each resulting "cluster" has a specific level or range of each variable. Thus, if we filter clusters having the same level of a variable (for example, age =<10), a correlation might be drawn from the remaining observations (where age>10) between some of the variables entered in the clustering process and the Wallet variable. These correlations can help us create the segments we are looking for.

   A question might arise as to why clustering was chosen instead of multivariate regression, where the demographic and in-house variables are the predictors and Wallet is the predicted variable. The reason is that we are looking to create segments with different Wallet distributions. We need to identify, in each of the final segments, the percentile that best represents the realistically attainable Wallet of all other customers in the concordant segment. Working with a multivariate regression will not give us the segments we require for the univariate stage of the analysis.

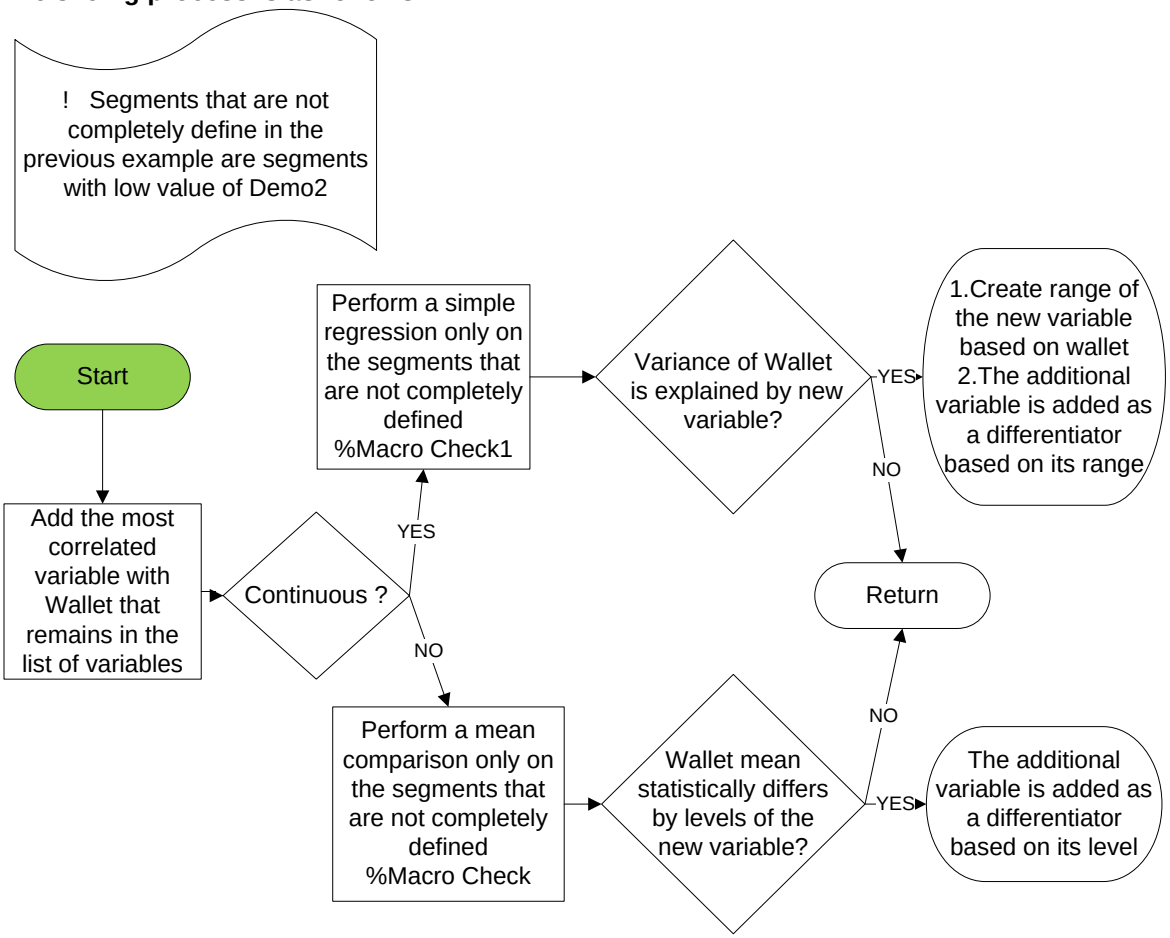
   Another method that can be used in Wallet estimation is tree analysis. A decision was taken not to use this method because while SAS ENTREPRISE MINER is a better product for tree analysis, it is not always available to all SAS users. Further, the clustering and slicing phases serviceably imitate the tree analysis process, eliminating the need for the more sophisticated product.

The clustering process is as follows:



ii. **Slicing:** When we are unable to create useful segments in the clustering phase, or if the segments display a high Wallet variance, we proceed to the slicing phase. The purpose of slicing is to identify any additional variables that will differentiate segments that were either not fully defined or were filtered in the clustering phase.

The slicing process is as follows:




```
%macro check (dsn =, var =, clus1=, clus2=);
/*dsn = dataset,
  var = new variable to be checked
  seg1 and seg2 are the segment numbers, which need further investigation
 TUKEY is a Pairwise comparison method */

 proc glm data= &dsn;
   class  &var;
   model sd_Wallet=&var;
   means &var/TUKEY alpha=0.05;
   where cluster not in (&clus1, &clus2);
 run;

 proc sort data = &dsn;
   by &var;
   where cluster not in (&clus1, &clus2); run;
   axis1 label= (height=2);
   axis2 label= (height=2);
 run;
```

```
proc boxplot data = &dsn;
   plot sd_Wallet*&var/ vaxis=axis1 haxis=axis2 cboxfill=TAN cboxes=BL;
   where cluster not in (&clus1, &clus2);
run;
%mend check;

%macro check1 (dsn =, var =, clus1=, clus2=);
/*dsn = dataset,
  var = Demographic variable to be checked
  seg1 and seg2 are the segment numbers which need further investigation*/

 proc reg data=&dsn;
    model sd_Wallet =&var/p;
    where cluster not in (&clus1, &clus2);
    output out=data_out p=yhat;
 run;

 proc gplot data=data_out;
    plot sd_Wallet* &var="*" yhat*&var ='#'/
    overlay legend /* add href if you want to visualize were the range
    should  be created href=value ch=purple */;
    where cluster not in (&clus1, &clus2);
 run;
 %mend check1;
```

4. SEGMENT EVALUATION

Evaluate the resulting segments with a final pairwise comparison using Tukey's method, in which each segment is compared with all other segments. One should aim for having a substantial number of pairs that are significantly different from one another.

5. WALLET ASSIGNMENT

In this stage, we focus on the Wallet distribution of each of the final segments in order to identify the most realistically attainable Wallet of the segment and assign it to all other customers in the same segment as their "potential Wallet" or Wallet assignment. We will account for a left- or right-skewed distribution by choosing a higher or lower top percentile.

DATA PREPARATION

While there are hundreds of demographic variables available, only a few offer genuine insight into a customer's Wallet size. In addition, since budgeting often restricts the size of the sample, there is a limit to the number of variables available to identify the final segments; hence, it is advisable to limit the pool of variables.

Variable reduction techniques were used to create an efficient list of demographic and in-house variables. Proc Varclus provides an expedient way to reduce the number of variables, as it clusters variables based on their correlation. Deciding which variables to keep in each cluster is a matter of evaluating them in terms of ease of implementation, information and variability, along with overall relevance to Wallet.

Additional data preparation steps include transformation of variables into numeric values, exclusion of outliers and the application of various imputation techniques for treating missing values.

ANALYSIS

So far, we have looked at the macro level of the Wallet estimation process. The following analysis offers a closer look, by way of examples, at Stages 3 through 5 of the Wallet estimation process; namely Segment Creation (Clustering and Slicing), Segment Evaluation and Wallet Assignment.

Note: This analysis was conducted on data collected through a survey, merged with demographic and in-house variables. Most of the data appearing in the graphs are kept standardized for proprietary reasons.

1. SEGMENT CREATION

Before using Proc Fastclus, standardize the data in order to put all variables on the same scale.

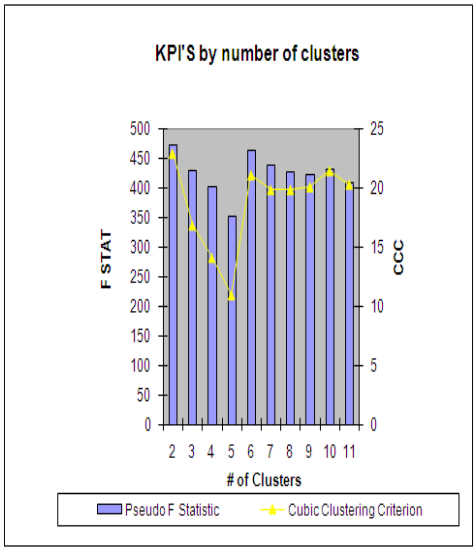
```
proc standard data=data2 out=Data3 mean=0 std=1;
   var &varlist;
run;
```

Minimize the effect of outliers on the clustering process by removing seeds with low-frequency clusters:

```
proc fastclus data=data3 outseed=mean1 maxc=20 maxiter=0 summary
mean=cluster_means;
   var &varlist;
run;

/*look at the ouput, find the numbers of observations in low frequency
clusters*/

data seed;
   set mean1;
   if _freq_>10; /*In this analysis, low frequency clusters have less than 10
observations*/
run;
```

Identifying the optimum number of clusters: The macro *cluster_* iterate on Proc Fastclus by increasing the number of clusters by one, and print the clustering results, in order to easily gather KPIs (Key Performance Indicators) and find the optimum number of clusters. The following KPIs are automatically generated by SAS: Pseudo F Statistic and Cubic Clustering Criterion.

```
%macro cluster_;
%do c=1 %to 12;
proc fastclus data=data3 maxclusters=&c summary seed=seed maxiter=200 distance
delete=20 least=2mean=cluster_means out=out_set1 outseed=mean;
   var &varlist;
run;
%end;
%mend cluster_;
%cluster_;
```



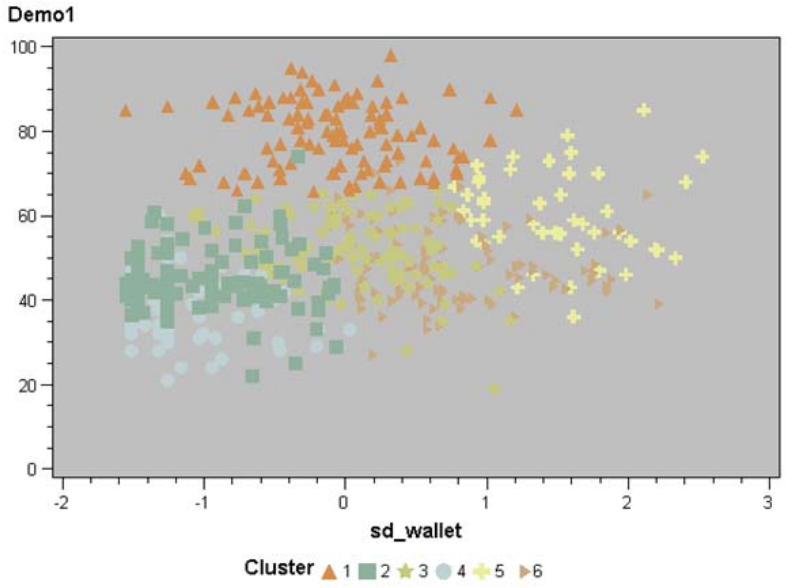
# of Clusters	Pseudo F Statistic	Cubic Clustering Criterion
2	472.1	22.856
3	429.76	16.798
4	401.44	14.067
5	352.5	10.919
6	463.32	21.029
7	437.79	19.805
8	427.7	19.797
9	422.55	20.042
10	432.15	21.387
11	409.88	20.253

Optimums appear at two and six clusters. Since we seek to identify segments with different Wallet distributions, six clusters is the best choice for Wallet segmentation. After choosing the optimum number

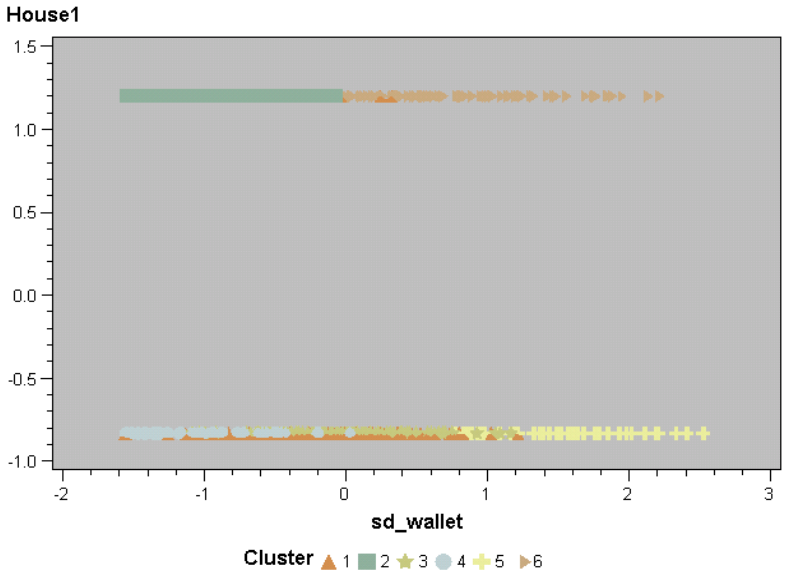
of clusters, we create one plot for each variable entered in the clustering process versus the Wallet variable. The following code runs Proc Fastclus with six clusters and the resulting data set is used in Proc Gplot to create the graphs:

```
proc fastclus data=data3 maxclusters=6 seed=seed maxiter=200 distance least=2
mean=cluster_mean1 out=out_set1 outseed=mean;
   var &varlist;
run;

proc gplot data=out_set1;
   plot Demo1*sd_Wallet=cluster/frame cframe=ligr legend=legend1
   vaxis=axis1 haxis=axis2;
run;

proc gplot data=out_set1;
   plot House1*sd_Wallet=cluster/frame cframe=ligr legend=legend1
   vaxis=axis1 haxis=axis2;
run;
```

PLOT1



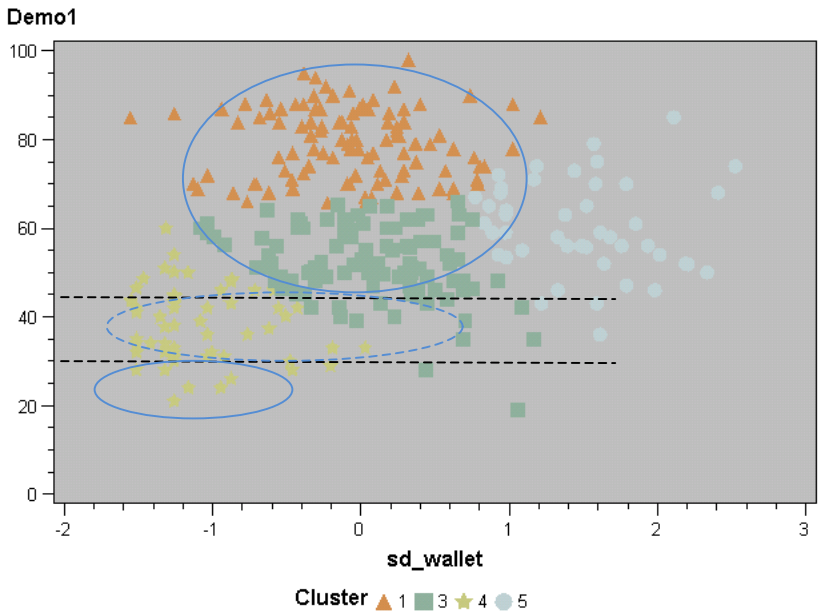
PLOT2



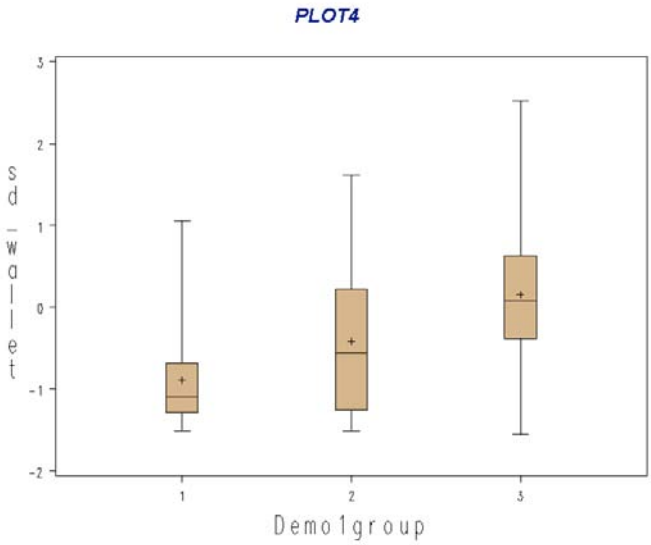
In Plot1, we see the formation of six different clusters. We can differentiate clusters 2 and 6 from clusters 1, 3, 4 and 5 by level of House1 (Plot2). If we filter Plot1 to capture only clusters in which values of House1 are negative, we see a resulting correlation between Wallet and Demo1 (see circles in Plot 3). Based on this information, we can designate Demo1 as a categorical variable called Demogroup1 (created in data=test2).

```
proc gplot data=outset1;
   plot Demo1*sd_Wallet=cluster/frame cframe=ligr legend=legend1
   vaxis=axis1 haxis=axis2;
   where cluster in (1,3,4,5);
run;
```

PLOT3




```
axis1 label=(height=2);
axis2 label=(height=2);

proc sort data=test2;
   by Demo1group;
run;

proc boxplot data=test2;
   plot sd_Wallet* Demo1Group/ vaxis=axis1 haxis=axis2 cboxfill=TAN cboxes=BL;
   where cluster not in (2, 6);
run;
```



Plot4 shows that Demo1Group=1 and Demo1Group=3 display different Wallet distributions. However, Demo1Group=2 has such variability with regard to Wallet that it requires further investigation. We proceed to the slicing phase in order to find the additional differentiator needed in Demo1Group=2.

Slicing of clusters where values of House1 are negative in Demo1Group=2 only: An additional variable from the list is added in order to find differentiation with regard to Wallet in clusters in which House1 is negative and Demo1Group=2. Since the additional variable is continuous, we proceed with the macro *%check1* to find an explanation for the variance of the Wallet in the undefined segment**.**

```
%check1(dsn =test2, var =house2,clus1=2,clus2=6)
```

(Note: In the macro, "`Where "DemoGroup=2"` was added)

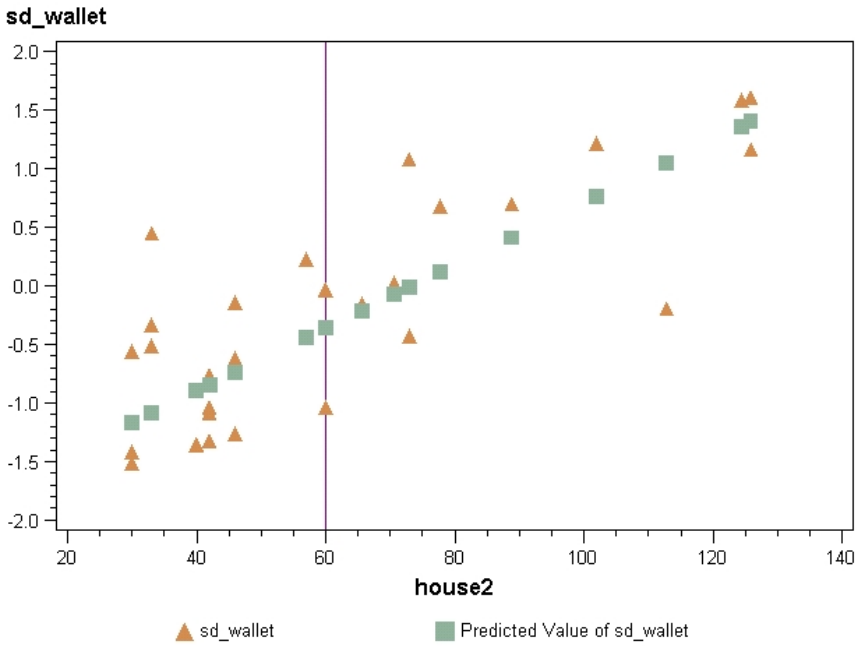
Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	21.22215	21.22215	62.88	<.0001
Error	33	11.13747	0.33750		
Corrected Total	34	32.35962			

Root MSE	0.58095	R-Square	0.6558
Dependent Mean	-0.41911	Adj R-Sq	0.6454
Coeff Var	-138.61487		

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	-1.96427	0.21820	-9.00	<.0001
house2	1	0.02677	0.00338	7.93	<.0001



House2 is a strong predictor for Wallet when House1 is negative and Demo1Group=2. Based on the plot we create a new variable, Demo1s2Group, in which low Wallet observations in Demo1group=2 (shown left of the reference line) are transferred to Demo1group=1, which displays a higher concentration of low Wallet values.

```
data test3;
   set test2;
   if (Demo1 <33 or (Demo1 >=33 and Demo1 <43 and House2 <60))
   then Demo1s2Group="1";
   if (Demo1 >=33 and Demo1 <43 and House2 >=60) then Demo1s2Group ="2";
   if Demo1 >=43 then Demo1s2Group ="3";
run;
```

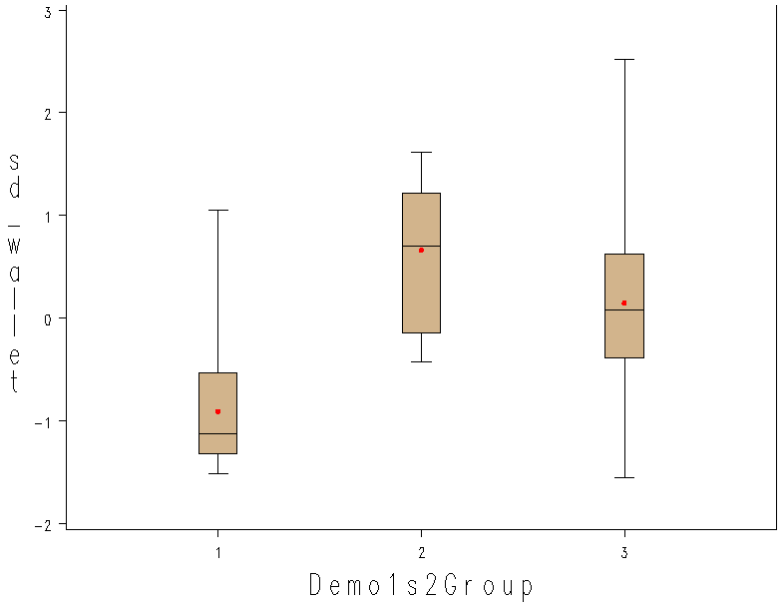
Since Demo1 was designated as a categorical variable, Demo1s2group, we use macro *%check* in order to assess whether all segments with negative values of House1 are significantly different from one another.

```
%check(dsn =test3, var =Demo1s2Group,clus1=2,clus2=6) ;
```

Comparisons significant at the 0.1 level are indicated by ***.

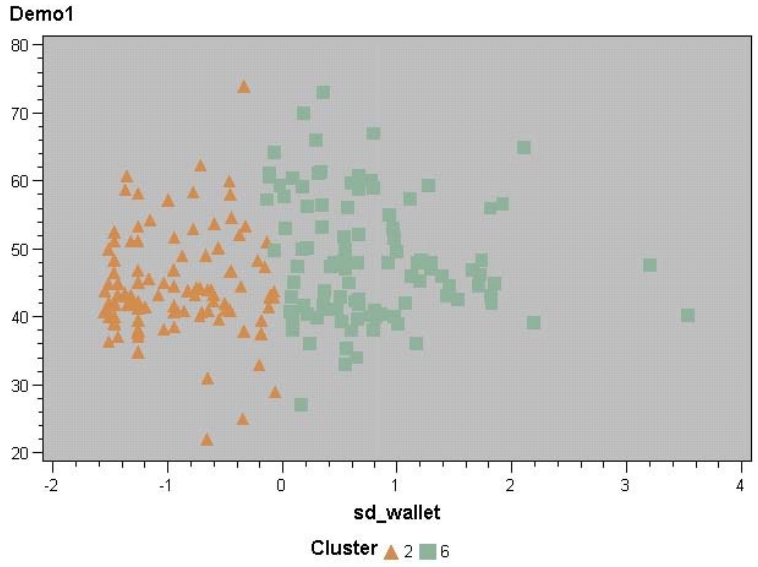
demo1s2group Comparison	Difference Between Means	Simultaneous 90% Confidence Limits		
2 - 3	0.51588	0.01619	1.01557	***
2 - 1	1.56971	1.02258	2.11685	***
3 - 2	-0.51588	-1.01557	-0.01619	***
3 - 1	1.05383	0.78911	1.31856	***
1 - 2	-1.56971	-2.11685	-1.02258	***
1 - 3	-1.05383	-1.31856	-0.78911	***

Based on the results of the mean comparison, we conclude that the mean of each segment is statistically different. The box plot below displays the differing Wallet tendencies of each segment.



Slicing stage for clusters with positive values of House1: Having defined clusters with negative values of House1, we proceed to define the remaining clusters, where values of House1 are positive.

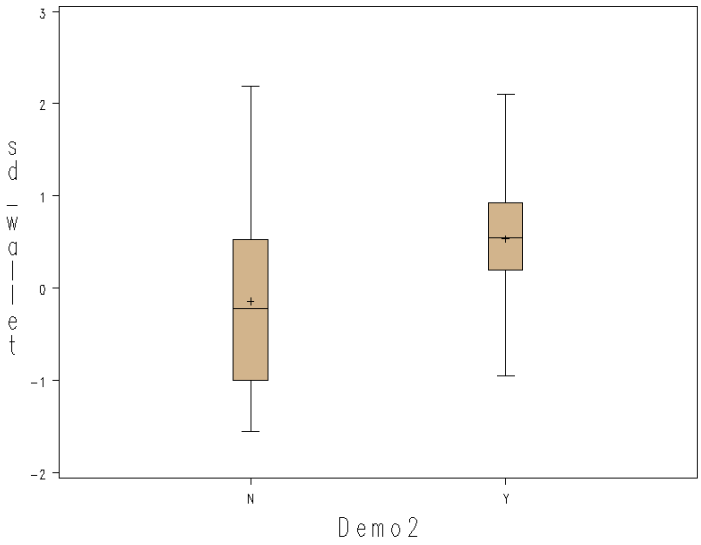
```
proc gplot data=test3;
   plot Demo1*sd_Wallet=cluster/frame cframe=ligr legend=legend1
   vaxis=axis1 haxis=axis2;
   where cluster in (2,6);
run;
```



It is apparent that Demo1 does not contribute to Wallet variance. Thus we proceed to the slicing phase and return to the pool of variables. After testing different variables, we find that the levels of Demo2 display a significant difference in Wallet in clusters with positive values of House1.

```
%check (dsn=test3, var=Demo2, clus1=2, clus2=6); quit;
```

(Note: In the macro, “cluster not in” was changed to “cluster in.”)



Means with the same letter are not significantly different.

Tukey Grouping	Mean	N	Demo2
A	0.5346	32	Y
B	-0.1430	79	N

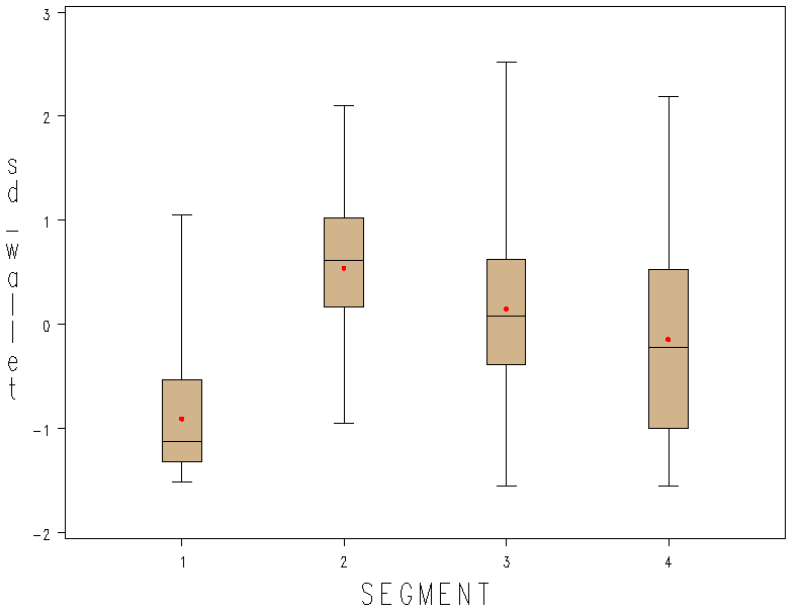
Since the clustering and slicing phases fully defined the segments, we proceed to the evaluation stage.

2. SEGMENT EVALUATION

At this stage, we gather all of the defined segments and perform an additional pair comparison with Proc Glm to determine whether any segments display similar Wallet distributions and thus need to be consolidated. In our analysis, two segments were found to be similar in terms of Wallet. We consolidated these into one segment. A final pair comparison showed that all means were significantly different from one another at the 0.1 level (see Proc Glm output below).

```
data segment;
   set test3;
    if house1<0 and (demo1 <33 or
    (demo1 >=33 and demo1 <43 and house2 <60)) then segment='1';
    if (house1<0 and demo1 >=33 and demo1 <43 and house2 >=60) or
    (house1>0 and demo2='Y') then segment='2';
    if house1<0 and demo1>=43 then segment='3';
    if house1>0 and demo2='N' then segment='4';run;
%check(dsn=segment, var=segment,clus1=0,clus2=0) ;
```

Comparisons significant at the 0.1 level are indicated by ***.				
SEGMENT Comparison	Difference Between Means	Simultaneous 90% Confidence Limits		
2 - 3	0.39171	0.08621	0.69722	***
2 - 4	0.68558	0.33364	1.03751	***
2 - 1	1.44791	1.04904	1.84679	***
3 - 2	-0.39171	-0.69722	-0.08621	***
3 - 4	0.29386	0.05285	0.53488	***
3 - 1	1.05620	0.75070	1.36171	***
4 - 2	-0.68558	-1.03751	-0.33364	***
4 - 3	-0.29386	-0.53488	-0.05285	***
4 - 1	0.76234	0.41040	1.11427	***
1 - 2	-1.44791	-1.84679	-1.04904	***
1 - 3	-1.05620	-1.36171	-0.75070	***
1 - 4	-0.76234	-1.11427	-0.41040	***



Having found the segments that differs in terms of Wallet, as well as the variables levels that define them, we proceed to the final stage, Wallet assignment.

3. WALLET ASSIGNMENT

The skewness value of the Wallet distribution in each segment, which indicates whether deviations from the mean are positive or negative, helps to determine which percentile will best represent the attainable Wallet of the segment. If for example a segment's Wallet distribution is normal i.e., showing a skewness value equal to or very close to zero, the 85th percentile is a good choice. This is preferable to choosing the maximum value, because with regard to Wallet estimation, it is best to avoid being overly optimistic. Choosing the 85th percentile eliminates most of the extreme values, making the estimate more probable.

If, on the other hand, the data points are asymmetric, we propose the following rule of thumb for choosing the most appropriate percentile of a segment:

```
proc univariate plot
   data=segment;
   var Wallet;
   by segment;
   output out=skew skewness=skewness;
run;
```

If the skewness value is close or equal to zero, choose the 85th percentile.
If the skewness value is less than zero, choose the 90th percentile.
If the skewness value is above zero (positive skew), choose the percentile P as follows:

```
P=round(100-|mean-median|/(2*|mean| ) )
```

In a positive skew value, the right tail is longer; also the mass of the distribution is concentrated on the left and has relatively few high values.

Once a percentile is chosen for each segment, the following code will assign a Wallet and Share of Wallet to each customer.

```
proc sort data=segment;
   by segment Wallet;
run;

proc univariate plot data=segment;
   var Wallet;
   by segment;
   output out=q pctlpre=p_ pctlpts=75,85,90;
run;

proc sql;
   create table percentile as select a.*,
   b.p_75,b.p_80,b.p_90,
   from segment as a left join q as b
   on a.segment=b.segment;
   quit;
data Wallet1;
   set percentile;
     if segment=1 then do;
       if internal_spend<=p_75 then Wallet_=p_75;
       else Wallet_=internal_spend;
     end;
     if segment=2 then do;
       if internal_spend<=p_90 then Wallet_=p_90;
       else Wallet_=internal_spend;
     end;     if segment=3 then do;
       if internal_spend<=p_80 then Wallet_=p_80;
       else Wallet_=internal_spend;
      end;
     if segment=4 then do;
       if internal_spend<=p_75 then Wallet_=p_75;
       else Wallet_=internal_spend;
      end;
run;

data Wallet2;
   set Wallet1;
    if internal_spend =Wallet_ then external_Wallet=0;
    else external_Wallet=Wallet_- internal_spend;
run;
data Wallet3;
   set Wallet2;
   share= internal_spend /Wallet_;
   where Wallet_ not in (0,.);
run;
```

```
data table ;
   set Wallet3;
   if (share>=0 and share<0.25) then share_='1';
   if (share>=0.25 and share<0.5 )then share_='2';
   if (share>=0.5 and share<0.75) then share_='3';
   if share>=0.75 then share_='4';
run;
```

CONCLUSION

Wallet and Share of Wallet estimation have valuable applications in marketing, throughout various industries, as they can be used to identify opportunities to increase (or maintain) Internal Spend and Share of Wallet.

Attrition Reduction: It can be deduced that customers with a low Share of Wallet, because they spend more money with other providers, have less brand loyalty and are susceptible to attrition. Identifying these customers makes it possible to target them with brand-building strategies designed to prevent attrition and convert External Spend to Internal Spend.

Retention: In segments where Share of Wallet is 75% and above, for example, it can be inferred that the customers are loyal to the company or brand. These customers can be targeted with retention tactics, and in general, be treated as customers who are satisfied with the products and services the company provides.

The insights to be gleaned from Wallet and Share of Wallet are well worth overcoming the challenges posed by the lack of ready availability of key data such as External Spend, which can be obtained through primary research.

REFERENCES

1. Saharon Rosset, Claudia Perlich, Bianca Zadrozny, Srujana Merugu, Sholom Weiss and Rick Lawrence (2005), "*Wallet Estimation Models*" Predictive Modeling Group, IBM T. J. Watson Research Center, Yorktown Heights, NY
2. Rajan Sambandam (2009), *Cluster Analysis Gets Complicated*. Reprinted with permission from the American Marketing Association (Marketing Research, Vol 15, No. 1, Spring 2003)
3. Mark E. Thompson (1999), "*The Science and Art of Market Segmentation Using PROC FASTCLUS"*, SUGI
4. Robert Adams(2008), Merck & Co., Inc., North Wales, PA,"*Box Plots in SAS®: UNIVARIATE, BOXPLOT, or GPLOT?*", NESUG

ACKNOWLEDGEMENTS

Professor David Madigan – Department of Statistics, Columbia University, New York

Mr. Drew Utman

Mr. Bilal Karriem – Independent Marketing consultant

CONTACT

Your comments and questions are welcomed. Please contact the author at:

Shira Guil Witelson
Email:Shira.Witelson@gmail.com