Partial Least Squares Regression using SAS and R

Avinash Mahesh Joshi

Innovation and Development

Mu Sigma Business Solutions

Bangalore

April 20, 2012

Abstract

The objective of this paper is to explore the reason for the discrepancies in the estimates calculated by R and SAS while running a partial least squares regression. Different options offered by both the softwares for performing the partial least squares regression (referred to as pls in the rest of the paper) are compared and the correct set of default options to be given to the pls algorithm are identified so that the estimates calculated by both the softwares reconcile.

Contents

1 INTRODUCTION

Partial least squares regression is an extension of the multiple linear regression model. In its simplest form, a linear model specifies the (linear) relationship between a dependent (response) variable Y, and a set of predictor variables, the $X's$, so that

$Y = b_0 + b_1X_1 + b_2X_2 + ... + b_pX_p$

In this equation b_0 is the regression coefficient for the intercept and the b_i values are the regression coefficients (for variables 1 through p) computed from the data.Usually, the variables in X and Y are centered by subtracting their means and scaled by dividing by their standard deviation.

1.1 CENTERING

Centering of the data means to subtract the mean of the column from each value of that column. Both SAS and R, while running pls, inherently perform the centering of the data by subtracting the means of the columns. Though SAS gives the option of not centering the data, R does not.

1.2 SCALING

Scaling of the data means to divide each value by the standard deviation of its column. Both SAS and R give the option to scale the data and also to not scale it. Meaning a manually scaled dataset by the user can be given and use the not scaling option in both the softwares.

Next section onwards, the estimates are printed for different conditions on scaling. The number of components is 5 in all the cases. The dataset used is dell - BHM with 9 independent variables and dependent variable being revenue.

2 Estimates of PLS by R and SAS without using the algorithm option for scaling.

These estimates are calculated, by using the option of not scaling the data in both the softwares. Basically the raw dataset is going into the algorithm and the algorithm is not scaling it.

R code for pls

```
> library("pls")
> pub <- read.csv("workpub.csv")
> pubPls = plsr(formula = rev_disc_pub ~ ., data = pub, ncomp = 5,
+     center = TRUE, scale = FALSE, method = "kernelpls", validation = "none")
> Estimates = round(as.vector(coef(pubPls, intercept = TRUE)),
+     2)
```

SAS code for pls

proc pls data=mmxs.scaledpub method=pls NOSCALE NFAC = 5 DETAILS; model rev_disc_pub = pub_incentive_cost print_spend_con_adstk3 dell_ds_ms_hpb_idc_pub_10 ple_print_spend cci total_domore_spend_adstk total_ple_ooh_adstk4 pub_seasonality sku_on_discount_pub/solution; quit;

Variable Name	Estimates in R	Estimates in SAS	OLS Estimates
Intercept	27162684.55	27162684.55	-8366642.16
pub_incentive_cost	16.34	16.34	9.60
print_spend_con_adstk3	3.78	3.78	7.19
dell_ds_ms_hpb_idc_pub_10	0	0	44792416.62
ple_print_spend	64.83	64.83	-16.20
cci	0.03	0.03	73391.26
total_domore_spend_adstk2	2.45	2.45	-0.66
total_ple_ooh_adstk4	80.24	80.24	54.02
pub_seasonality	0	0	23747331.77
sku_on_discount_pub	1.15	1.15	346075.98

Table 1: Estimates for PLS regression using R and SAS on the raw dataset and scale = FALSE

The estimates exactly match.

3 Estimates of PLS by R and SAS using the algorithm option for scaling.

These estimates are calculated, by using the option scaling the data in both the softwares. Basically the raw dataset is going into the algorithm and the algorithm is scaling it first before doing the partial least squares regression.

R code for pls

```
> library("pls")
> pub <- read.csv("workpub.csv")
> scaledPubPls = plsr(formula = rev_disc_pub ~ ., data = pub, ncomp = 5,
+     scale = TRUE, method = "kernelpls", validation = "none")
> Estimates = round(as.vector(coef(scaledPubPls, intercept = TRUE)),
+     2)
```

SAS code for pls

proc pls data=mmxs.pub method=pls NFAC = 5 DETAILS; model rev_disc_pub = pub_incentive_cost print_spend_con_adstk3 dell_ds_ms_hpb_idc_pub_10 ple_print_spend cci total_domore_spend_adstk2 total_ple_ooh_adstk4 pub_seasonality sku_on_discount_pub/solution; quit;

Variable Name	Estimates in R	Estimates in SAS	OLS estimates
Intercept	-8490337.3	-8490337.3	186816654.02
pub_incentive_cost	3570596.4	9.66	9.60
print_spend_con_adstk3	3838074.3	7.2	7.19
dell_ds_ms_hpb_idc_pub_10	3338897.8	47443447.61	44792416.62
ple_print_spend	-605225.1	-14.99	-16.20
cci	604803.6	60004.07	73391.26
total_domore_spend_adstk2	-113622.2	-0.17	-0.66
total_ple_ooh_adstk4	2608721.9	50.9	54.02
pub_seasonality	7669597.2	23770189.23	23747331.77
sku_on_discount_pub	17047188.5	345310.96	346075.98

Table 2: Estimates for PLS regression using R and SAS on the raw dataset and scale = TRUE

The estimates do not match.

4 Estimates of PLS by R and SAS by manually scaling the data.

These estimates are calculated, by first manually scaling the data as per user's choice (in the below case, the scaling factor is standard deviation of the column) and then, using the option to not scale in the algorithm.

R code for pls

```
> library("pls")
> pub <- read.csv("workpub.csv")
> cenScaledPub <- as.data.frame(scale(pub))
> cenScaledPubPls = plsr(formula = rev_disc_pub ~ ., data = cenScaledPub,
+     ncomp = 5, scale = FALSE, method = "kernelpls", validation = "none")
> Estimates = round(as.vector(coef(cenScaledPubPls, intercept = TRUE)),
+     2)
```

SAS code for pls

proc pls data=mmxs.scaledpub method=pls NOSCALE NFAC = 5 DETAILS; model rev_disc_pub = pub_incentive_cost print_spend_con_adstk3 dell_ds_ms_hpb_idc_pub_10 ple_print_spend cci total_domore_spend_adstk total_ple_ooh_adstk4 pub_seasonality sku_on_discount_pub/solution; quit;

Variable Name	Estimates in R	Estimates in SAS	OLS estimates
Intercept	-0.22	-0.22	186816654.02
pub_incentive_cost	0.09	0.09	9.60
print_spend_con_adstk3	0.1	0.1	7.19
dell_ds_ms_hpb_idc_pub_10	0.09	0.09	44792416.62
ple_print_spend	-0.02	-0.02	-16.20
cci	0.02	0.02	73391.26
total_domore_spend_adstk2	0.00	0.00	-0.66
total_ple_ooh_adstk4	0.07	0.07	54.02
pub_seasonality	0.2	0.20	23747331.77
sku_on_discount_pub	0.44	0.44	346075.98

Table 3: Estimates for PLS regression using R and SAS on the manually scaled dataset and scale = FALSE

The estimates exactly match.

5 Comparison of estimates for pls in R between algorithm scaling and manual scaling

Following table compares the estimates obtained by giving the raw dataset to the code and asking the code to scale it with the estimates obtained by manually scaling the dataset and then giving it to the algorithm asking it not to scale.

R code for pls(algorithm scaling)

```
> pub <- read.csv("workpub.csv")
> scaledPubPls = plsr(formula = rev_disc_pub ~ ., data = pub, ncomp = 5,
+     scale = TRUE, method = "kernelpls", validation = "none")
> Estimates = round(as.vector(coef(scaledPubPls, intercept = TRUE)),
+     2)
```

R code for pls(manual scaling)

```
> pub <- read.csv("workpub.csv")
> cenScaledPub <- as.data.frame(scale(pub))
> cenScaledPubPls = plsr(formula = rev_disc_pub ~ ., data = cenScaledPub,
+     ncomp = 5, scale = FALSE, method = "kernelpls", validation = "none")
> Estimates = round(as.vector(coef(cenScaledPubPls, intercept = TRUE)),
+     2)
```

Variables Name	Estimates in R on dataset(Scale = TRUE)	Estimates in R on manually scaled dataset
Intercept	-8490337.3	-0.22
pub_incentive_cost	3570596.4	0.09
print_spend_con_adstk3	3838074.3	0.1
dell_ds_ms_hpb_idc_pub_10	3338897.8	0.09
ple_print_spend	-605225.1	-0.02
cci	604803.6	0.02
total_domore_spend_adstk2	-113622.2	0
total_ple_ooh_adstk4	2608721.9	0.07
sku_on_discount_pub	17047188.5	0.44

Table 4: Estimates for pls regression in R for algorithmic scaling and manual scaling

Even these do not match suggesting that the scaling option is manipulating not just the dataset but also the algorithm hence should be avoided.

6 CONCLUSION

The estimate results do not match for both the softwares when the scaling option is turned on for the respective softwares. There is a difference in the way in which both the softwares deal with scaling. The best option is to manually scale the data by subtracting the mean and dividing by the standard deviation and give this standardized data to the pls algorithm. If the dataset is input in its standardized form then it is immaterial whether you choose the scaling option or not as the data will be remain the same on scaling again as the mean is '0' and standard deviation is '1'. But it is recommended that the data is manually scaled and then in the algorithm, the option not to scale is chosen.

Hence the option to not scale should be the default option and the user should be asked to center and scale the data as per user's choice and then give that dataset as an input to the algorithm in R. But now the estimates are in the standardized form (z scores form), these need to be convertedback to original form. This will result in same results by both the softwares.