Partial Least-Squares Splines :

Presentation of the

PLSS package (Version 20.61)

Jean-François Durand

Laboratoire de Probabilités et Statistique,
Montpellier II University, France

E-Mail: support@jf-durand-pls.com
Web site: www.jf-durand-pls.com

Data Mining and Regression Tools : IASC-Procida 2007-1

I. Short presentation of the functions

1. Downloads

R-package ↦ http://cran.r-project.org/
Source files ↦ http://www.jf-durand-pls.com/ProgramSources.html

2. Four main R-functions

FUNCTION	TITLE	MENU
PLSL PLSS plscalibration	Linear PLS PLS Splines Linear PLS for Near Infrared Spectroscopy	TITLE (0 to exit) 1: Bivariate Analysis for Spline Inputs (*1:) 2: Cross Validation 3: Generalized Cross Validation 4: A look at graphics for exploration and modeling (*4:) 5: Selection of the predictors (Remove/Add) (*5:) Selection:
MAPLSS	Multivariate Additive PLS Splines	TITLE (0 to exit) 1: CV/GCV for pure main effects models (mandatory) 2: Automatic selection of interactions 3: Validation, ANOVA plots, External prediction (*3:) 4: Pruning main effects and interactions Selection:

In "PLSL" : (*1:) "Bivariate Analysis"
(*4:) "Conversational PLS"
In "plscalibration" : (*1:) "Looking at the spectra"
(*4:) "Conversational PLS"
(*5:) not available

In all functions:
(*3:)(*4:) External prediction available only if "Xtest" input is not missing

3. Computational inputs

Common to all functions		
Input	*Description*	*Default*
X="matrix"	learning sample predictors (mandatory).	
Y="matrix"	learning sample responses (mandatory).	
Xtest="matrix"	test sample predictors.	missing (*)
Ytest="matrix"	test sample responses.	missing
D="vector"	stat. weights of observations, if D=1 equidistributed.	1
StandX="bool"	if =T, X is D-standardized, if =F, only D-centered.	T
StandY="bool"	idem for Y.	T
A="num"	number of PLS components.	1
prop="num"	proportion of left/predicted observations in CV.	0.1
GCV="num"	positive GCV parameter, if 0, CV is processed.	1

plscalibration		
Input	*Description*	*Default*
spect="vector"	integer vector of the training sample spectra.	1:nrow(X)
spectest="vector"	integer vector of the test sample spectra.	1:nrow(Xtest)
byscale="num"	integer value, 1/*byscale* means that only 1 column of X (wavelength) among *byscale* is accounted for.	1

MAPLSS		
Input	*Description*	*Default*
interaction="vector"	integer vector of predictors for possible interactions.	1:ncol(X)
PRESSprop="num"	threshold for accepting or not one interaction.	0.2

$\mapsto$ All matrices must have non null dimnames.

$\mapsto$ (*) Xtest=X in plscalibration.

4. Inputs for graphics

Common to all functions		
Input	*Description*	*Default*
bgpar="string"	background color name. Use colors( ) to choose.	"lightblue"
askpar="boolean"	if T, press ENTER to see next plot.	T
ptypar, cexpar, pchpar	usual R-plot parameters: pty, cex, and pch.	"s", 0.7, "*"
typedata="boolean"	if T, residual plot with the name of the observations, if F, their number only.	T
qual="vector"	nrow(X) integers giving the color of the observations.	missing
names.qual="vector"	vector of *qual* level names to use in a legend sub-plot.	missing
titlepar="boolean"	if T, put a title in multiple plots.	T
colpar="num"	initialize the first color to be used.	1
matrow="num"	number of rows in a matrix of plots.	1
matcol="num"	number of columns in a matrix of plots.	1

"Looking at the spectra" in plscalibration		
Input	*Description*	*Default*
steps="boolean"	if T, displays the spectra one-by-one in *stepsorder* order.	F
stepsorder="boolean"	if T, the increasing order.	T
titlestring="string"	title of plots	"Training sample"
matrow, matcol	matrix dimension of the 3 plots: mean spectrum, first and second derivatives.	1, 3

3-D perspective plots in plscalibration and MAPLSS		
Input	*Description*	*Default*
thetapar, phipar, rpar	(θ, ϕ, r) eye spherical coordinates.	-60, 30,10

5. Spline inputs in PLSS and MAPLSS

Degree		
Input	*Description*	*Default*
degree="vector"	vector of ncol(X) integers, the degrees to transform the predictors, if degree=d, all predictors have the same degree, say d.	1

Knots		
Rapid strategy without a priori information		
Input	*Description*	*Default*
knots="vector"	vector of ncol(X) integers, the numbers of knots for the predictors, if knots=k, all predictors have the same number, say k.	0
equiknots="vector"	vector of ncol(X) T/F values. If T, equally spaced knots, if F, knots located at the *knots* - quantiles of the predictor. If equiknots=F, all predictors have knots at *knots* - quantiles.	F
Individual choice for each predictor		
Input	*Description*	*Default*
listknots="list"	list of ncol(X) vectors of the location of knots, if *listknots* is missing, then *knots* and *equiknots* are available.	missing

↦ Default values provide, for all predictors, degree=1 and knots=0 (empty set of knots) leading to a linear PLS model.

↦ The first item in the PLSS menu, "Bivariate Analysis for Spline Inputs", allows direct on-line graphical selection of the spline inputs.

II. Some examples of R-commands

Download the text file of these commands and the data sets involved, from the web-pages

Source Files and Data sets for the courses.

First input : the predictor matrix X *observations* × *variables*.
Second input : the response matrix Y *observations* × *variables*.

1. Linear PLS on Cornell Data

```
PLSL(cornell[,1:7],cornell[,8,drop=F],cexpar=0.8,matrow=3,matcol=3)
```

2. Linear PLS calibration on TECATOR Data

```
plscalibration(meatX,meatY[,1,drop=F],spect=1:172,Ytest= meatY[,1,drop=F],
spectest=173:215,matrow=3,matcol=1,titlestring= "Absorbance Calibration
Sample",askpar=T)
```

Training samples 1:172, test samples 173:215

```
plscalibration(meatX,meatY[,1,drop=F],spect=1:172,Ytest= meatY[,1,drop=F],
spectest=173:215,matrow=3,matcol=1,titlestring= "Calibration Sample",
steps=T,steporder=F,askpar=T)
```

Absorbance spectra are step by step sequentially displayed in descending order.

---

3. Pure additive model on Orange Juice Data

The building model stage

A campaign of PLSS tries allows to construct the additive model.

```
try0<-PLSS(juicX,juicY[,1,drop=F],matrow=3,matcol=3)
```

Default spline parameters for all predictors: degree=1, knots=0.

```
try1<-PLSS(juicX,juicY[,1,drop=F],degree=2,knots=2)
```

Spline parameters for all predictors: degree = 2, 2 knots at quantiles.

```
try2<-PLSS(juicX,juicY[,1,drop=F],degree=2,knots=2,equiknots=T)
```

Spline parameters for all predictors: degree = 2, 2 equally spaced knots, by default knots at quantiles (equiknots=F).

```
localknots<-list(c(400,1600),c(10,20,40),c(10,40),c(2.5,5),c(160,400),
c(40,110), c(4,11,30),c(400,1700),c(100,300,500),c(600,2600))
try3<-PLSS(juicX,juicY[,1,drop=F],degree=2,listknots=localknots)
```

Spline parameters : degree = 2, individual knots for each predictor.

A look at the PLSS outputs

Results are stored in try3, a list of 3 objects:

try3$Xvariables, try3$degree and try3$listknots.

try3$Xvariables is the boolean indicator of the retained predictors in the selection of the main effects (Menu 5), the two others characterize the spline inputs.

```
try3
$Xvariables
[1] FALSE FALSE FALSE FALSE TRUE TRUE FALSE TRUE TRUE
[10] FALSE
$degree
[1] 2 2 2 2 2 2 2 2 2 2
$listknots
$listknots[[1]] [1] 400 1600
$listknots[[2]] [1] 10 20 40
$listknots[[3]] [1] 10 40
$listknots[[4]] [1] 2.5 5.0
$listknots[[5]] [1] 160 400
$listknots[[6]] [1] 40 110
$listknots[[7]] [1] 4 11 30
$listknots[[8]] [1] 400 1700
$listknots[[9]] [1] 100 300 500
$listknots[[10]] [1] 600 2600
```

Results of the final retained model

```
PLSS(juicX[,try3$Xvariables],juicY[,1,drop=F],degree=2,listknots=
localknots[try3$Xvariables])
```

Retained predictors :

```
dimnames(juicX)[[2]][try3$Xvariables]
```

```
[1] "Ca" "Mg" "SO4" "HCO3"
```

Optimal dimension : 2

degree : 2

knots location :

```
localknots[try3$Xvariables]
[[1]] [1] 160 400
[[2]] [1] 40 110
[[3]] [1] 400 1700
[[4]] [1] 100 300 500
```

Optimal PRESS : "PRESS(0.1,2)=0.1157" with "prop=0.1", the proportion of observations out.

Optimal GCV : "GCV(1.8,2)=0.1144", with "GCV=1.8" the tuning GCV parameter.

---

4. MAPLSS on the orange juice data: no interaction

Here are only searched eventual interactions between the main predictors 5, 6, 8 and 9, selected in the pure additive PLSS model.

```
MAPLSS(juicX,juicY[,1,drop=F],degree=2,listknots=localknots, GCV=1.8,
interaction=c(5,6,8,9))
```

Tuning parameter for the GCV criterion. When 0, the Cross-Validation is processed, the default value is 1. One has to previously calibrate that GCV parameter in the PLSS campaign, that is, to find a dimension and a GCV-criterion value as close as possible respectively to the dimension and the PRESS-criterion.

Vector of predictors for the candidate interactions, the default, 1:ncol(X), means that all bivariate interactions are candidate: ncol(X)(ncol(X)-1)/2.*

No interaction is retained. Notice that testing all possible 45 interactions between the 10 predictors, gives the same result:

```
MAPLSS(juicX,juicY[,1,drop=F],degree=2,listknots=localknots, GCV=1.8)
```

---

5. Testing MAPLSS on simulated data

The function f constructs (X,Y) data bases. X, n by p, is a matrix of p n-dimensional covariate vectors uniformly generated in the unit hypercube. Y, n by 1, is the "signal + noise" response.

f to generate "Y=f(X)+noise" data		
Input	*Description*	*Default*
signal="string"	the signal, a function of the X columns.	""
n="integer"	number of observations.	100
p="integer"	number of predictors (X columns).	1
stdv="numerical"	standard deviation of the normal *N(0, stdv)* noise.	1
seedpar="numerical"	the seed for generating numbers at random.	20
Output	*Description*	*Default*
$X="matrix"	$n \times p$ training sample sample *U[0, 1]* predictors.	100 *U[0, 1]*
$Y="matrix"	$n \times 1$ training sample response, f(X)+noise.	100 *N(0, 1)*
$Ysignal="matrix"	$n \times 1$ signal f(X)	

```
f2<-"10*sin(pi*X[,1]*X[,2])+20*(X[,3]-0.5)^2+10*X[,4]+5*X[,5]"
```

Through the following R-commands, the response does not depend on the 5 last predictors (among 10) that contribute as pure noise. Two experiments with f to generate training and test samples and allow to compute the Mean Squared Error on test data:

```
experiment1<-f(f2,n=100,p=10,seedpar=20)
Xtrain<-experiment1$X
Ytrain<-experiment1$Y
experiment2<-f(f2,n=100,p=10,seedpar=120)
Xtest<-experiment2$X
signaltest<-experiment2$Ysignal
```

```
MAPLSS(X=Xtrain,Y=Ytrain,Xtest=Xtest,Ytest=signaltest,degree=c(1,1,2,1,1,1,
1,1,1,1),listknots=list(0.5,0.5,0.5,NULL,NULL,NULL,NULL,NULL,NULL,NULL))
```

*The interaction detected is the true X1*X2, with a GCV gain relative to the pure additive main effects GCV that amounts to 56.72%.*

```
 2.2: Incorporating interactions step by step :
------------------------------------------------
 Reference : Main effects GCV(1,3) = 0.114791
------------------------------------------------
candidate  1  :  X1*X2 ACCEPTED. at 20 % relative GCV gain
      i j A    GCV         %rel.GCVgain
X1*X2 1 2 6 0.04967867        56.72
candidate  2  :  X2*X4 NOT ACCEPTED. at 20 % relative GCV gain
      i j A    GCV          %rel.GCVgain
X2*X4 2 4 6 0.04581933         7.77
Continue anyway exploring (y/n)?1: n
------------------------------------------------
Range of the transformed predictors in descending order:
   X4    X1*X2      X3      X5      X1      X2     X10      X7     X9
 1.75962  1.31994 0.95828 0.95438 0.79419 0.76246 0.14731 0.08212 0.0694
   X8      X6
 0.06561 0.05034
```

One can now prune the 7 lowest ANOVA terms and retain only the first 4. Because Xtest is not missing, external prediction is present in the third item of the menu.

```
---------------------------------------
Validation, ANOVA plots, External prediction, 0 to exit

1: Validation
2: ANOVA plots
3: External prediction

Selection : 3
---------------------------------------
selected main effects
X3 X4 X5
selected interactions
   X1*X2
Validation of the model with the test sample according to 10 dimensions
Mean Squared Error of the response(s) according to the dimension
      1    2     3     4    5     6    7     8     9     10
MSE 5.27 2.004 0.77 0.609 0.49 0.473 0.392 0.383 0.38  0.383
Choose the dimension (<= 10 )1: 8
```

Since $Ytest_i = f_2(Xtest_i)$, $i = 1, \ldots, n$

Mean Squared Error:

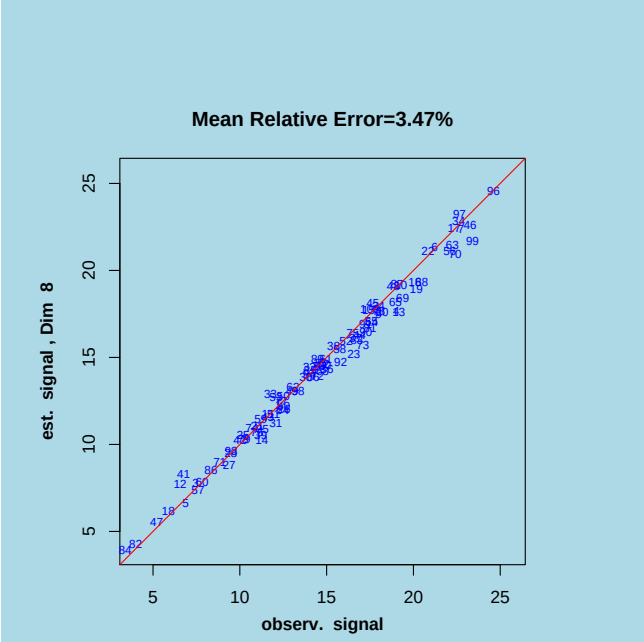
$$\text{MSE(M)} = \frac{1}{n}\sum_{i=1}^{n}(Ytest_i - \hat{Y}(M)_i)^2$$

$$MSE(8) = 0.383$$

Mean Relative Error:

$$\text{MRE(M)} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{Ytest_i - \hat{Y}(M)_i}{Ytest_i}\right|$$

$$MRE(8) = 0.0347$$



6. MAPLSS on the "chem" data: one interaction

We skip the preliminary PLSL linear model giving PRESS(0.02,2)=0.5709.

First, we select and save the spline inputs for a PLSS main effects model:

```
try1=PLSS(chem[,1:10],chem[,11,drop=F])
> try1
$Xvariables
 [1] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
$degree
 x1  x2  x3  x4  x5  x6  x7  x8  x9 x10
  1   1   0   1   1   1   1   1   1   3
$listknots
$listknots$x1
[1] 0.6
$listknots$x2
NULL
$listknots$x3
[1] 0.2
$listknots$x4
NULL
$listknots$x5
[1] 0.3
$listknots$x6
NULL
$listknots$x7
[1] 0.3
$listknots$x8
NULL
$listknots$x9
NULL
$listknots$x10
NULL
```

Second, we validate a PLSS main effects model by leave-one-out CV,

```
PLSS(chem[,1:10],chem[,11,drop=F],degree=try1$degree,listknots=try1$listknots,
prop=0.02)
```

that provides poor CV results, $PRESS(0.02, 3) = 0.4835$ *and* $GCV(1, 3) = 0.44716$.

At that time, two possibilities:
first, a campaign of PLSS tries to improve spline inputs giving a better validation,
second, look for capturing eventual interactions by MAPLSS.

```
try2=MAPLSS(chem[,1:10],chem[,11,drop=F],degree=try1$degree,listknots=
try1$listknots,askpar=T)
```

Final model: $x_4 * x_{10}$ interaction detected leading to PRESS(0.02,6)=0.0569.

Ordered ANOVA terms according to the range of the spline transformation:

```
   x4*x10      x8       x2       x5       x7
 5.07837 0.70734 0.59963 0.55541 0.45304
```

Summary of leave-one-out CV results (prop=0.02):

linear	main effects	main effects+interactions
PLSL	PLSS	MAPLSS
PRESS(0.02,2)=0.5709	PRESS(0.02,3)=0.4835	PRESS(0.02,6)=0.0569