Partial Least Squares Regression (part II)

Predictive Modeling & Statistical Learning

Gaston Sanchez

PLS Regression

PLS Regression equation in terms of the original predictors:

$$\begin{aligned}\mathbf{y} &= d_1\mathbf{z_1} + d_2\mathbf{z_2} + \mathbf{e}\\ &= d_1\mathbf{X}\mathbf{w_1} + d_2\mathbf{X_1}\mathbf{w_2} + \mathbf{e}\\ &= \mathbf{X}(d_1 w_1^* + d_2 w_2^*) + \mathbf{e}\\ &= b_1\mathbf{x_1} + b_2\mathbf{x_2} + \cdots + b_p\mathbf{x_p} + \mathbf{e}\end{aligned}$$

Properties of PLS Regression

Properties

- $\mathbf{z_h}^\top \mathbf{z_l} = 0, \quad l > h$
- $\mathbf{w_h}^\top \mathbf{p_h} = 1$
- $\mathbf{w_h}^\top \mathbf{X_l}^\top = 0, \quad l \geq h$
- $\mathbf{w_h}^\top \mathbf{p_l} = 0, \quad l > h$
- $\mathbf{w_h}^\top \mathbf{w_l} = 0, \quad l > h$
- $\mathbf{z_h}^\top \mathbf{X_l} = 0, \quad l \geq h$
- $\mathbf{X_h} = \mathbf{X} \prod_{j=1}^{p} (\mathbf{I} - \mathbf{w_j} \mathbf{p_j}^\top), \quad h \geq 1$

Modified Weights $\mathbf{w}^*_\mathrm{h}$

We know that $\mathbf{z_h} = \mathbf{X_{h-1}w_h}$

$\mathbf{z_h}$ can also be expressed as $\mathbf{z_h} = \mathbf{Xw^*_h}$

$$\mathbf{w^*_h} = \prod_{k=1}^{h-1}(\mathbf{I} - \mathbf{w_k p_k^\intercal})\mathbf{w_h}$$

Modified Weights $\mathbf{w}_{\mathbf{h}}^{*}$

In fact,

$$\mathbf{W}_{\mathbf{h}}^{*} = \mathbf{W}_{\mathbf{h}}(\mathbf{P}_{\mathbf{h}}^{\mathsf{T}}\mathbf{W}_{\mathbf{h}})^{-1}$$

$$\mathbf{Z}_{\mathbf{h}} = \mathbf{X}\mathbf{W}_{\mathbf{h}}(\mathbf{P}_{\mathbf{h}}^{\mathsf{T}}\mathbf{W}_{\mathbf{h}})^{-1}$$

Decomposition

The matrices of PLS components $\mathbf{Z}$ and loadings $\mathbf{P}$ can be used to decompose $\mathbf{X}$ as:

$$\mathbf{X} = \mathbf{Z}\mathbf{P}^{\mathsf{T}}$$

It can be shown that:

$$\hat{\boldsymbol{\beta}}_{OLS} = \sum_{h=1}^{p} d_h \mathbf{w}_{\mathbf{h}}^{*}$$

$$\hat{\mathbf{y}}_{OLS} = d_1 \mathbf{z_1} + d_2 \mathbf{z_2} + \cdots + d_p \mathbf{z_p}$$

What is PLSR doing?

Why PLS is worth it?

- The answer is stability of predictors.
- PLS keeps the number of variables as low as possible.
- In PLS, components are selected that give maximal reduction in the covariance $\mathbf{X}^\mathsf{T}\mathbf{y}$ of the data.
- In that sense PLS will give the minimum number of variables that is necessary.
- The PLS regression is based on the SVD of $\mathbf{X}^\mathsf{T}\mathbf{y}$

Some Insights

The first PLS component has the form $\mathbf{z} = \mathbf{X}\mathbf{w}$
Under the hood, the PLS regression involves **Tucker** criterion:

$$\arg\max_{\mathbf{w}} \left\{ cov^2(\mathbf{y}, \mathbf{X}\mathbf{w}) \right\}$$

What is this criterion doing?

Some Insights

Recall that the covariance can be expressed as:

$$cov(\mathbf{y}, \mathbf{z}) = cor(\mathbf{y}, \mathbf{z}) \sqrt{var(\mathbf{y})} \sqrt{var(\mathbf{z})}$$

thus:

$$cov^2(\mathbf{y}, \mathbf{z}) = cor^2(\mathbf{y}, \mathbf{z}) \, var(\mathbf{y}) \, var(\mathbf{z})$$

Some Insights

What does PLSR optimize?

$$\arg\max_{\mathbf{w}} \left\{ cov^2(\mathbf{y}, \mathbf{X}\mathbf{w}) \right\}$$

is equivalent to:

$$\arg\max_{\mathbf{w}} \left\{ cor^2(\mathbf{y}, \mathbf{z})\ var(\mathbf{y})\ var(\mathbf{z})) \right\}$$

PLSR is a compromise between the multiple regression of $\mathbf{y}$ on $\mathbf{X}$, and the PCA of $\mathbf{X}$

Some Insights

Tucker's criterion $cov^2(\mathbf{y}, \mathbf{Xw})$ is a compromise between:

- maximizing correlation $cor(\mathbf{z}, \mathbf{y})$ (OLS regression)
- maximizing variance of PLS components $var(\mathbf{Xw})$

Advantages

- PLSR is not based on any optimization criterion.
- Rather it is based on an interative algorithm (which converges).
- However, it turns out that the PLS-solution is equivalent to the SVD of $\mathbf{X}^\mathsf{T}\mathbf{y}$

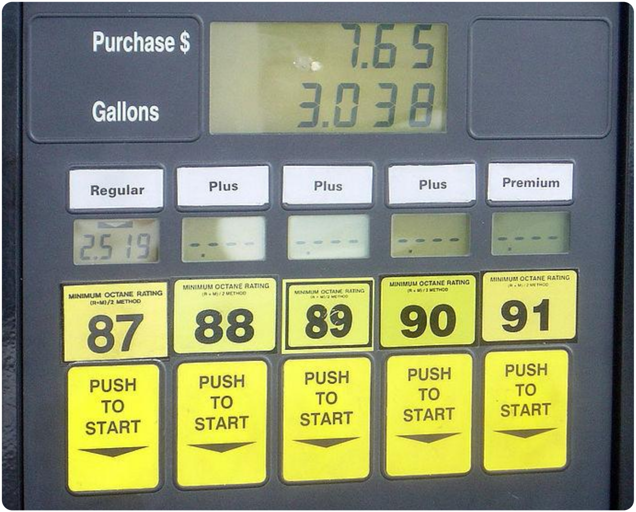
Advantages of PLS Regression

Advantages

- Simplicity in its algorithm
- No need to invert any matrix
- No need to diagonalize any matrix
- You just need to compute simple regressions
- In other words, you just need inner products
- Missing data is allowed (but you need to modify the algorithm)
- Easily extendable to the multivariate case of various responses
- Handles cases where we have more predictors than observations ($p >> n$)

Example:
Gasoline Data

Gasoline Octane Ratings

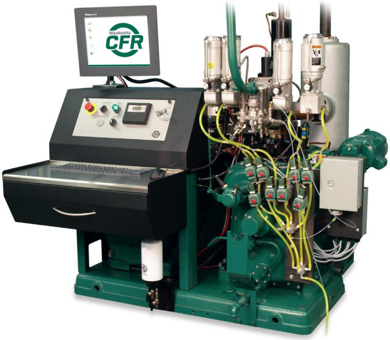


https://commons.wikimedia.org/wiki/File:Gas_Station_Pump_Five_Octane_Ratings.jpg

Predicting Octane Number

- Predicting octane number of a gasoline from the NIR (Near Infra Red) spectrum of gasolines.
- The **octane number**, or octane rating, is a standard measure of the performance of an engine or aviation fuel.
- The higher the octane number, the more compression the fuel can withstand before detonating (igniting).
- Fuels with a higher octane rating are used in high performance gasoline engines that require higher compression ratios.

Research Octane Number (RON)



http://www.waukeshacfr.com/f1-f2/

The most common type of octane rating worldwide is the Research Octane Number (RON). RON is determined by running the fuel in a test engine with a variable compression ratio under controlled conditions, and comparing the results with those for mixtures of iso-octane and n-heptane.

Dataset `gasoline.txt`

- 60 gasolines, 402 variables
- Response Y: octane number
- Predictors $X_1, \dots, X_{401}$: NIR spectrum frequencies (900nm-1700nm)
- As you can tell: $p >> n$
- We'll use the first 50 gasolines as the training set
- The remaining gasolines (last 10) will be used as test set

Dataset gasoline.txt

Data file gasoline.txt in the data/ folder of the github repo

```r
gasoline <- read.table("gasoline.txt", header = TRUE)
```

```r
dim(gasoline)

## [1]  60 402
```

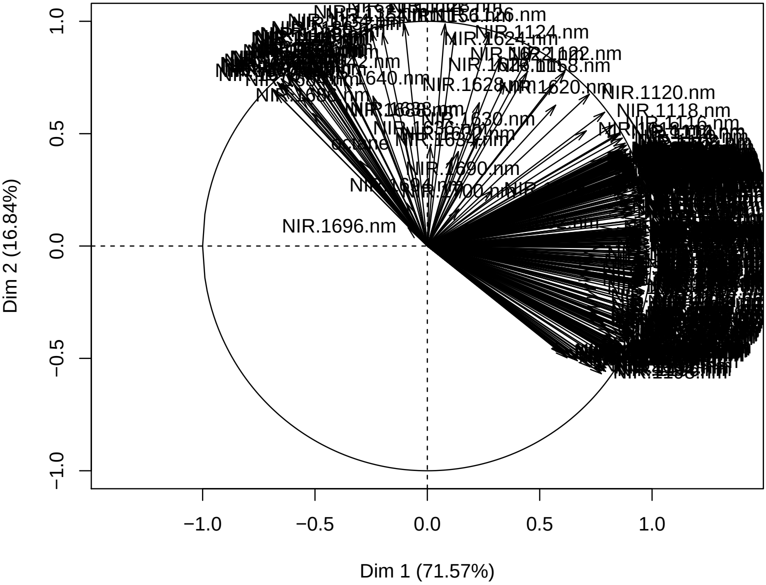
Dataset `gasoline.txt`

First few rows of data:

```
  octane NIR.900.nm NIR.902.nm NIR.904.nm NIR.906.nm
1  85.30  -0.050193  -0.045903  -0.042187  -0.037177
2  85.25  -0.044227  -0.039602  -0.035673  -0.030911
3  88.45  -0.046867  -0.041260  -0.036979  -0.031458
4  83.40  -0.046705  -0.042240  -0.038561  -0.034513
5  87.90  -0.050859  -0.045145  -0.041025  -0.036357
6  85.50  -0.048094  -0.042739  -0.038812  -0.034017
7  88.90  -0.049906  -0.044558  -0.040543  -0.035716
```

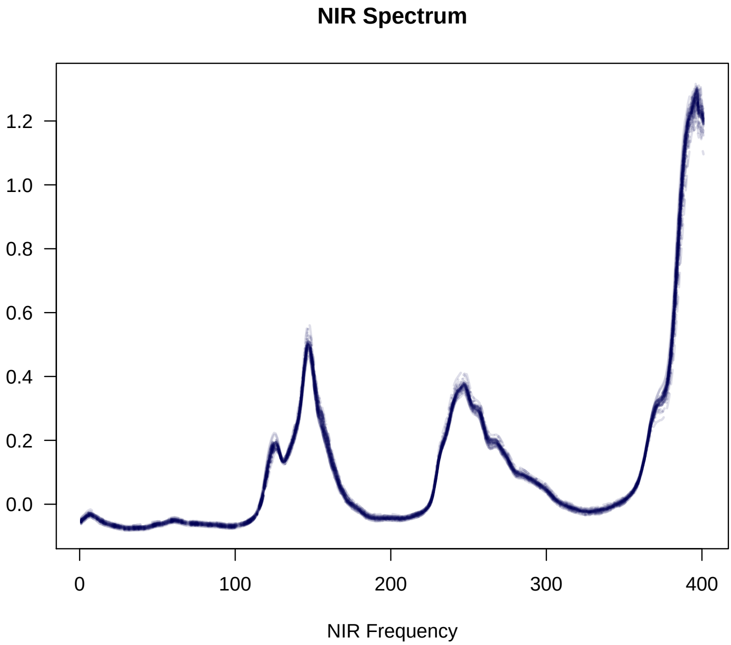
- First column `octane` is the response.
- Rest of columns are predictors.

Variables factor map (PCA)



Example: gasoline.txt

```r
# response
octane <- gasoline[,1]

# predictors
NIR <- gasoline[,2:ncol(gasoline)]

# training and test sets
train <- 1:50
test <- 51:60
```


NIR Spectrum
1.2
1.0
0.8
0.6
0.4
0.2
0.0
0
100
200
300
400
NIR Frequency

```r
corrs <- cor(NIR, octane)
summary(corrs)
```

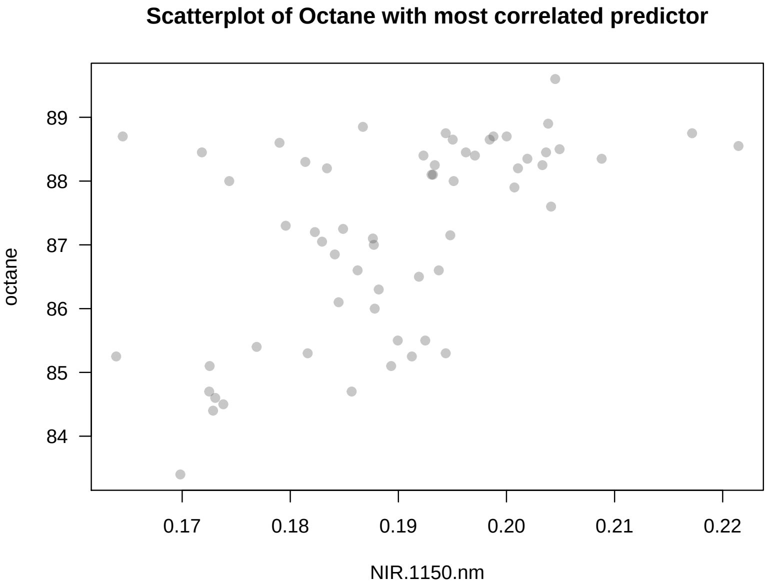
```
       V1
 Min.   :-0.90362
 1st Qu.:-0.38877
 Median :-0.19437
 Mean   :-0.18578
 3rd Qu.:-0.05055
 Max.   : 0.56396
```

```r
which.max(corrs)
```

```
[1] 126
```

```r
corrs[which.max(corrs)]
```

```
[1] 0.5639595
```


Scatterplot of Octane with most correlated predictor
octane
89
88
87
86
85
84
0.17
0.18
0.19
0.20
0.21
0.22
NIR.1150.nm

Out of curiosity let's try OLS with lm()

```r
# OLS regression attempt
gas_train <- gasoline[1:50, ]
reg <- lm(octane ~ ., data = gas_train)
summary(reg)
```

```
Residuals:
ALL 50 residuals are 0: no residual degrees of freedom!

Coefficients: (352 not defined because of singularities)
             Estimate Std. Error t value Pr(>|t|)
(Intercept)    136.74         NA      NA       NA
NIR.900.nm   -2276.35         NA      NA       NA
NIR.902.nm     144.74         NA      NA       NA
...

Residual standard error: NaN on 0 degrees of freedom
Multiple R-squared:      1, Adjusted R-squared:    NaN
F-statistic:   NaN on 49 and 0 DF,  p-value: NA
```

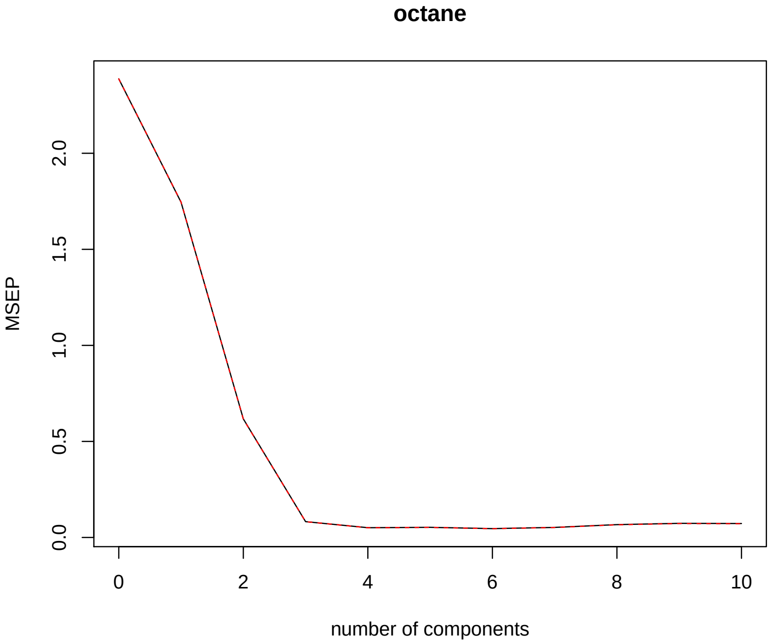
Partial Least Squares Regression

```
library(pls)

set.seed(1)
pls1 <- plsr(octane ~ ., ncomp = 10, data = gasoline, subset = train,
             scale = TRUE, validation = "LOO")

pls1


## Partial least squares regression , fitted with the kernel algorithm.
## Cross-validated using 50 leave-one-out segments.
## Call:
## plsr(formula = octane ~ ., ncomp = 10, data = gasoline, subset = train,
```

Summarized Output from plsr()

```r
summary(pls1)
```

```
## Data:  X dimension: 50 401
##  Y dimension: 50 1
## Fit method: kernelpls
## Number of components considered: 10
##
## VALIDATION: RMSEP
## Cross-validated using 50 leave-one-out segments.
##        (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps
## CV           1.545    1.321   0.7857   0.2869   0.2254   0.2295   0.2145   0.2287
## adjCV        1.545    1.322   0.7848   0.2866   0.2251   0.2287   0.2141   0.2279
##        8 comps  9 comps  10 comps
## CV      0.2586   0.2710    0.2695
## adjCV   0.2567   0.2692    0.2676
##
## TRAINING: % variance explained
##         1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps  8 comps
## X         64.31    85.24    95.79    97.22    97.59    98.19    98.61    98.74
## octane    31.59    79.29    97.13    98.49    98.91    99.01    99.10    99.37
##         9 comps  10 comps
## X         99.10     99.25
## octane    99.46     99.57
```

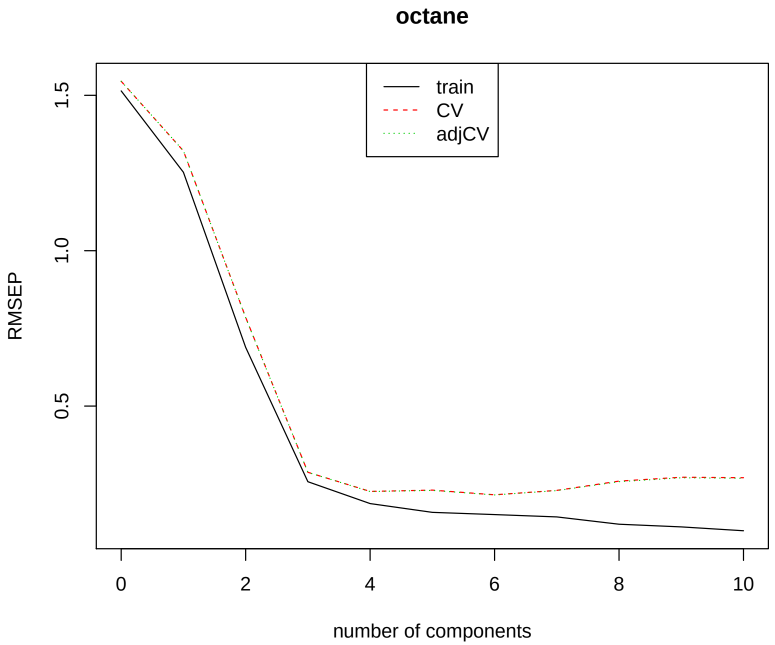

octane
MSEP
0.0
0.5
1.0
1.5
2.0
0
2
4
6
8
10
number of components

Test MSEs

```
##        ncomp   MSE_test
## [1,]       1 1.61005872
## [2,]       2 0.56881957
## [3,]       3 0.19325159
## [4,]       4 0.03332153
## [5,]       5 0.19678268
## [6,]       6 0.08161284
## [7,]       7 0.10074208
## [8,]       8 0.26969233
```

which minimum test MSE?

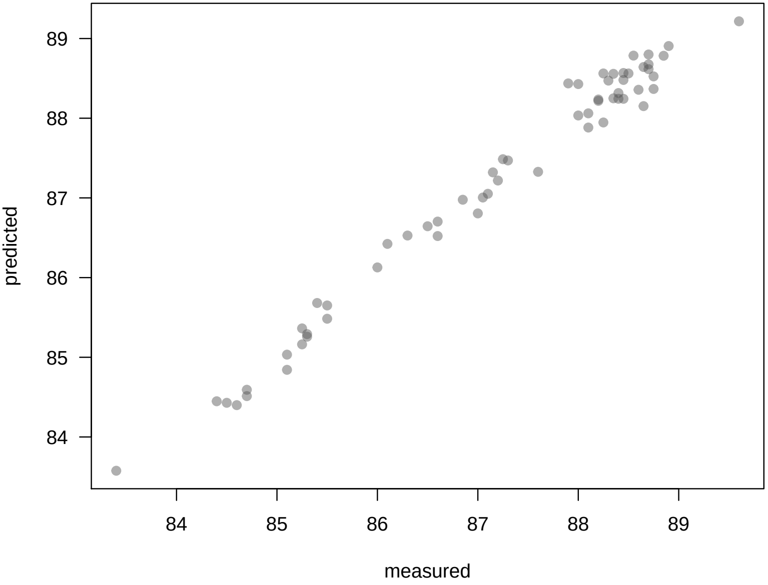
```
## [1] 4
```


octane
train
CV
adjCV
RMSEP
1.5
1.0
0.5
0
2
4
6
8
10
number of components

Final PLS Regression

```
pls_fit <- plsr(octane ~ ., ncomp = 4, data = gasoline, scale =
summary(pls_fit)

## Data:  X dimension: 60 401
##  Y dimension: 60 1
## Fit method: kernelpls
## Number of components considered: 4
## TRAINING: % variance explained
##         1 comps  2 comps  3 comps  4 comps
## X         64.97    83.51    93.72    96.33
## octane    30.54    79.79    97.73    98.27
```

Observed and predicted values (4 PLS comps)



References

- **Modern Multivariate Statistical Techniques** by Julian Izenman (2008). *Chapter 5, sec 6: Biased Regression Methods*. Springer.
- **Linear Models with R** by Julian Faraway (2015). *Chapter 11: Shrinkage Methods*. CRC Press.
- **Some theoretical aspects of partial least squares regression** by Inge Helland (2001). *Chemometrics and Intelligent Laboratory Systems, 58, 97-107*.
- **Partial Least Squares Regression and Statistical Models** by Inge Helland (1990). *Scandinavian Journal of Statistics. Vol. 17, No. 2. p. 97-114*.

References (French Literature)

- **La Regression PLS: Theorie et Pratique** by Michel Tenenhaus (1998). Editions, Technip.
- **Probabilites, analyse des donnees et statistique** by Gilbert Saporta (2011). *Chapter 17: La regression multiple et le modele lineaire general*. Editions Technip, Paris.