# MAE 5776

# ANÁLISE MULTIVARIADA

Júlia M Pavan Soler

pavan@ime.usp.br

1º Sem/2020 - IME

# Análise Multivariada

Já vimos ☺

- Análise Descritiva Multivariada
- Elipsóides de Dispersão e de Confiança, MANOVA
- Metodologias Clássicas: Foco na obtenção de vetores reducionistas

$$Y_{n\times p} = \left(Y_{ij}\right) \in \Re^{n\times p} \qquad \Re^{p} \rightarrow \Re^{m}$$

$n > p$
Observações Independentes

- ✓ ACP, ACoP, AC, AF
- ✓ AG
- ✓ AD, ACC
- ✓ Soluções Duais ($\Re^{nxp}$, $\Re^{pxp}$, $\Re^{nxn}$), Representações Biplot
- ✓ PCR (Regressão via CP), PLS (Regressão via MQ Parciais)

Integração de Bancos de Dados: **AD, ACC, PLS**
Diagrama de Caminhos ou Grafos

$\Rightarrow$ n << p (*big p*)
$\Rightarrow$ Observações dependentes

Métodos via Fatoração de Matrizes e via Modelos de Regressão!

# PLSR e CCA: Integração de BD

Escore das obs | Cargas das var | Escore das obs | Cargas das var

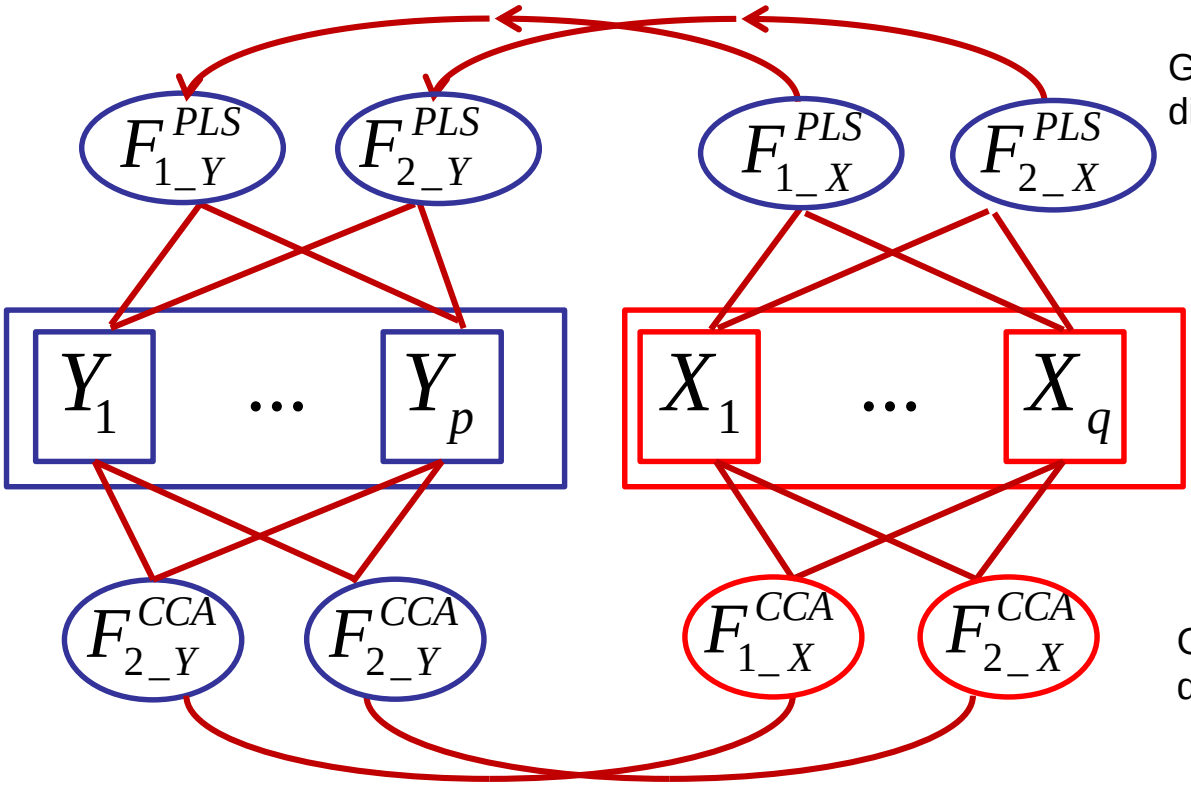
Fatoração de Matrizes

$$Y_{n\times p} = U_Y\ \Lambda_Y\ V_Y{}' = F_Y\ W_Y \qquad X_{n\times q} = U_X\ \Lambda_X\ V_X{}' = F_X\ W_X$$

PLS $\left[Cov\left(b'Y;a'X\right)\right]^2 = Var\left(b'Y\right)\left[Corr\left(b'Y;a'X\right)\right]^2 Var\left(a'X\right)$ CCA

**PLS**

**CCA**




Grafo direcionado

Grafo Não-direcionado

# PCR e PLSR - Exemplo

Dados disponíveis no pacote `pls` do R:

- **gasoline**: $\left[ Y_{60\times 1} \quad X_{60\times 401} \right]$

```
> data("gasoline")
> ??gasoline
starting httpd help server ... done
> names(gasoline)
[1] "octane" "NIR"
> str(gasoline)
'data.frame':   60 obs. of  2 variables:
 $ octane: num  85.3 85.2 88.5 ... Y: número de octane
 $ NIR   : 'AsIs' num [1:60, 1:401] -0.0502 -0.0442 -0.0469 ... X: espectros NIR
```

```
gasTrain <- gasoline[1:50, ] #treinamento
gasTest <- gasoline[51:60, ] #teste
```

**Y: gasoline$octane**

| | |
|---|---|
| 1 | 85.30 |
| 2 | 85.25 |
| 3 | 88.45 |
| ... | ... |
| 58 | 86.60 |
| 59 | 89.60 |
| 60 | 87.10 |

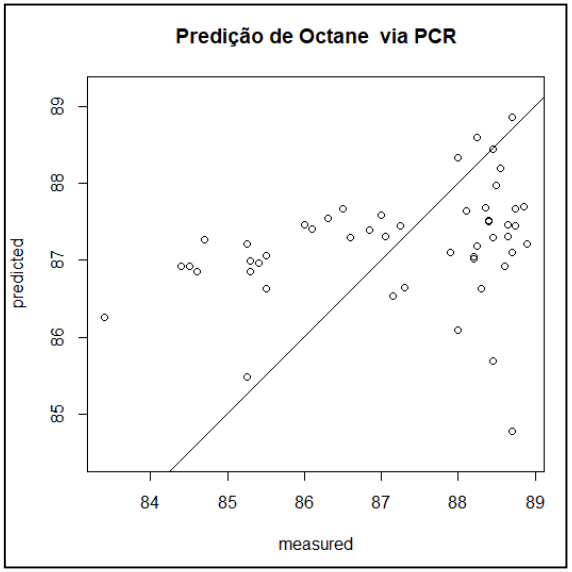
**X: gasoline$NIR**

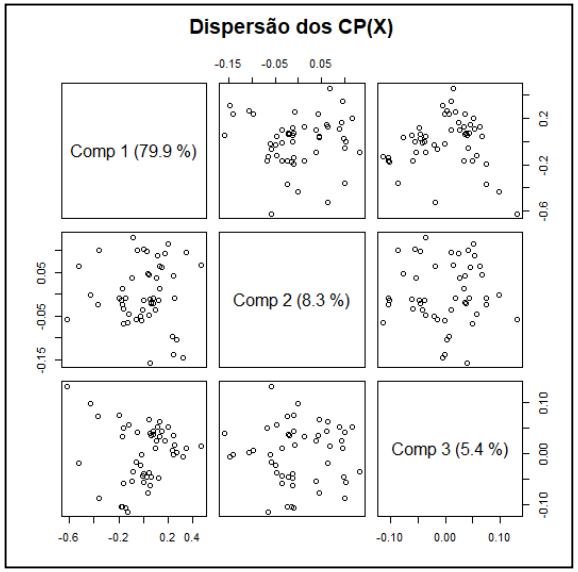
| | 900nm | 902nm | ... | 1698nm | 1700nm |
|---|---|---|---|---|---|
| 1 | -0.050193 | -0.045903 | ... | 1.245913 | 1.221135 |
| 2 | -0.044227 | -0.039602 | ... | 1.227985 | 1.198851 |
| 3 | -0.046867 | -0.041260 | ... | 1.230321 | 1.208742 |
| ... | | | ... | | |
| 58 | -0.053693 | -0.048020 | ... | 1.191871 | 1.150779 |
| 59 | -0.056311 | -0.051231 | ... | 1.175997 | 1.154696 |
| 60 | -0.058805 | -0.053311 | ... | 1.154355 | 1.163959 |

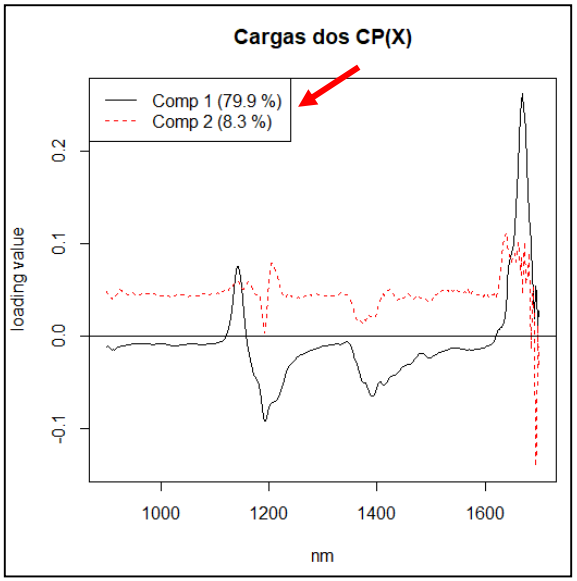
Explorar os recursos do `pls do R`

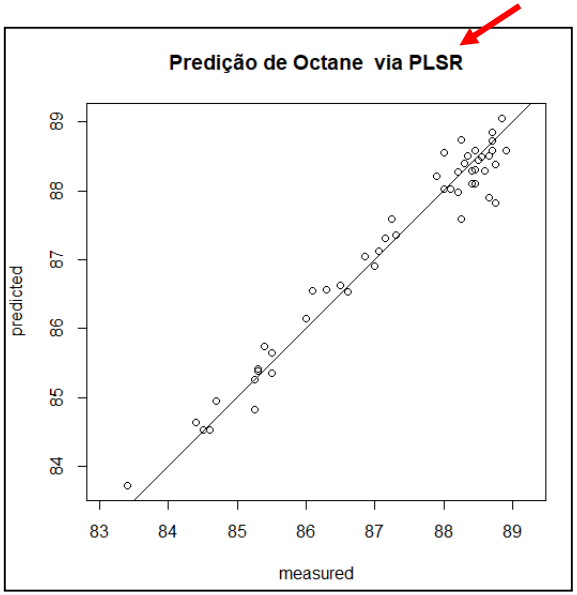
# PCR e PLSR - Gasoline

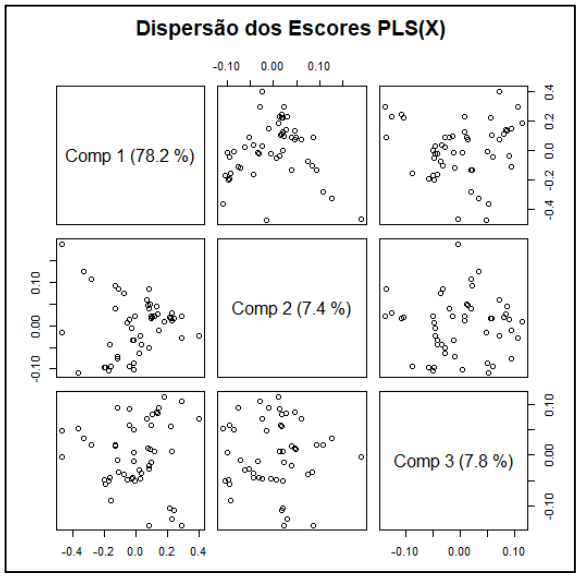
$[Y_{60\times1} \quad X_{60\times401}]$

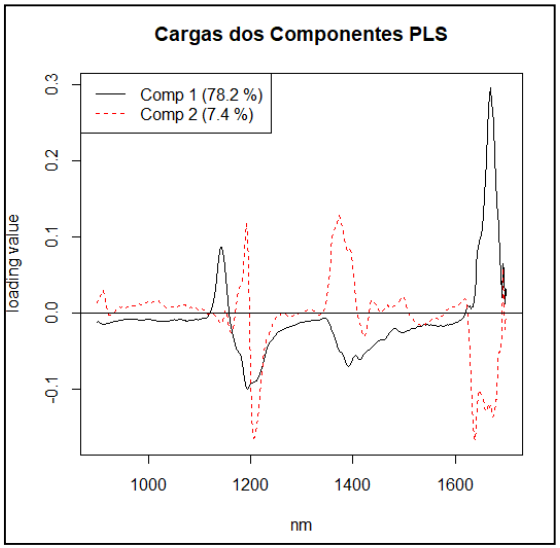


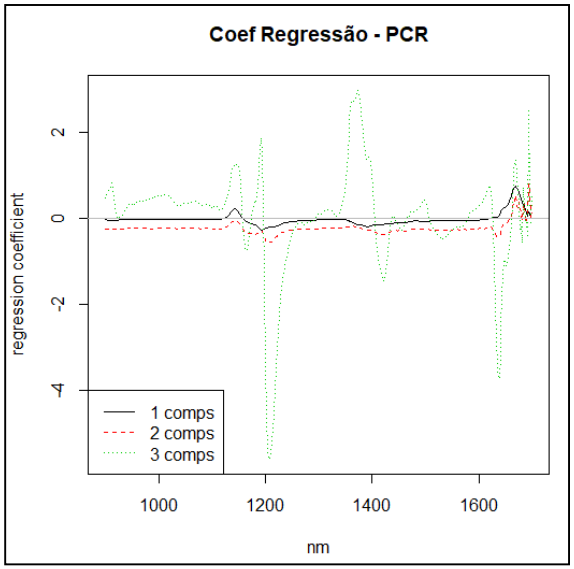



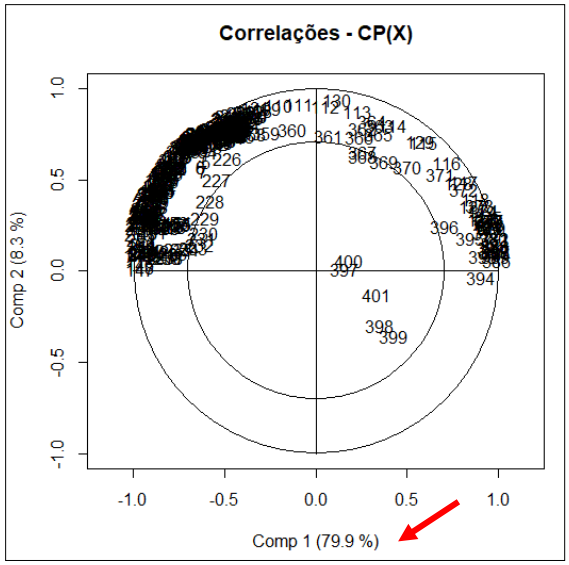



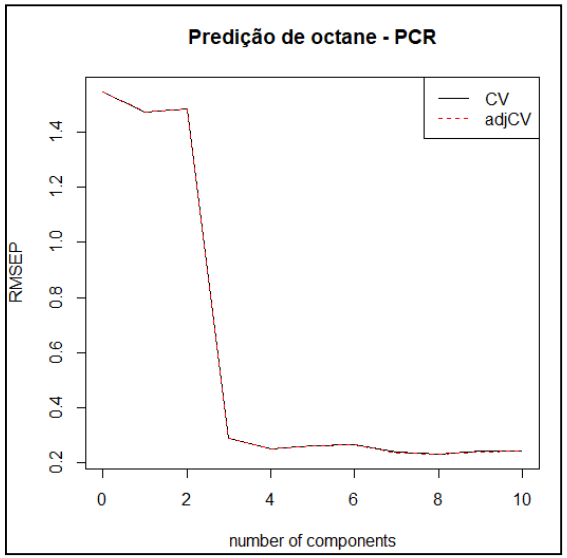



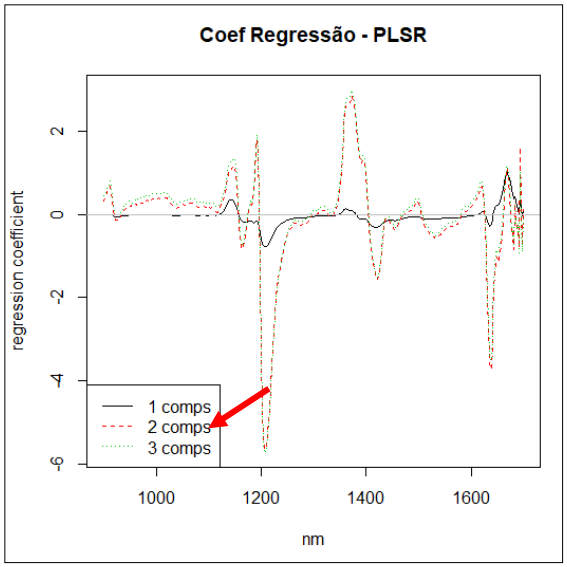



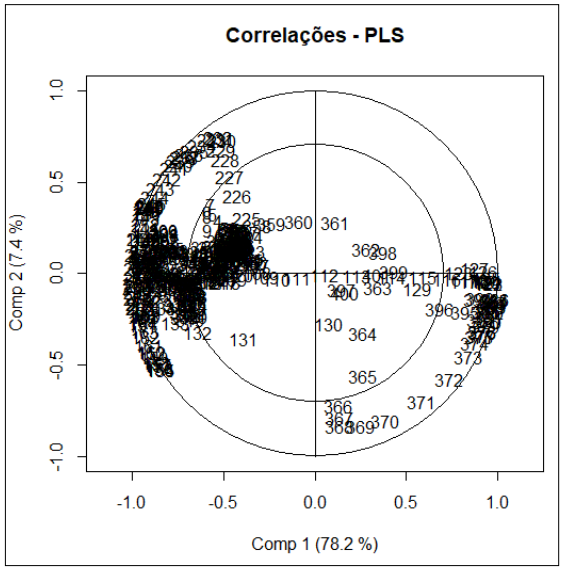
# PCR e PLSR - Gasoline

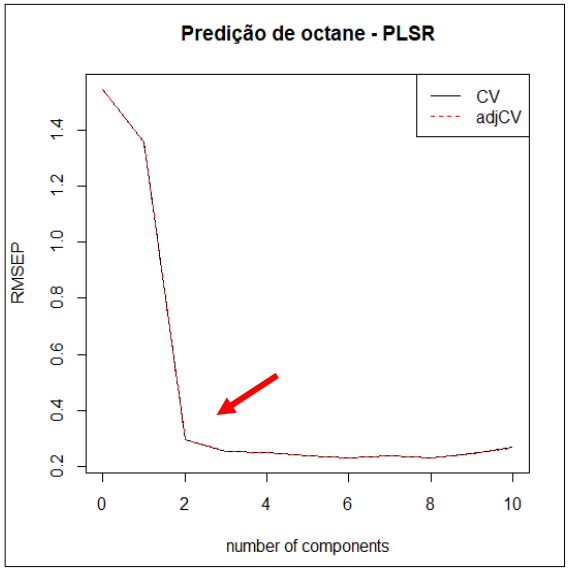


# PLSR - Exemplo

Dados disponíveis no pacote `pls` do R:

- **`oliveoil`**: $\left[ Y_{16\times 6} \quad X_{16\times 5} \right]$

**Y: oliveoil$sensory**

| | yellow | green | brown | glossy | transp | syrup |
|---|---|---|---|---|---|---|
| G1 | 21.4 | 73.4 | 10.1 | 79.7 | 75.2 | 50.3 |
| G2 | 23.4 | 66.3 | 9.8 | 77.8 | 68.7 | 51.7 |
| G3 | 32.7 | 53.5 | 8.7 | 82.3 | 83.2 | 45.4 |
| G4 | 30.2 | 58.3 | 12.2 | 81.1 | 77.1 | 47.8 |
| G5 | 51.8 | 32.5 | 8.0 | 72.4 | 65.3 | 46.5 |
| I1 | 40.7 | 42.9 | 20.1 | 67.7 | 63.5 | 52.2 |
| I2 | 53.8 | 30.4 | 11.5 | 77.8 | 77.3 | 45.2 |
| I3 | 26.4 | 66.5 | 14.2 | 78.7 | 74.6 | 51.8 |
| I4 | 65.7 | 12.1 | 10.3 | 81.6 | 79.6 | 48.3 |
| I5 | 45.0 | 31.9 | 28.4 | 75.7 | 72.9 | 52.8 |
| S1 | 70.9 | 12.2 | 10.8 | 87.7 | 88.1 | 44.5 |
| S2 | 73.5 | 9.7 | 8.3 | 89.9 | 89.7 | 42.3 |
| S3 | 68.1 | 12.0 | 10.8 | 78.4 | 75.1 | 46.4 |
| S4 | 67.6 | 13.9 | 11.9 | 84.6 | 83.8 | 48.5 |
| S5 | 71.4 | 10.6 | 10.8 | 88.1 | 88.5 | 46.7 |
| S6 | 71.4 | 10.0 | 11.4 | 89.5 | 88.5 | 47.2 |

**X: oliveoil$chemical**

| | Acidity | Peroxide | K232 | K270 | DK |
|---|---|---|---|---|---|
| G1 | 0.73 | 12.70 | 1.900 | 0.1390 | 0.003 |
| G2 | 0.19 | 12.30 | 1.678 | 0.1160 | -0.004 |
| G3 | 0.26 | 10.30 | 1.629 | 0.1160 | -0.005 |
| G4 | 0.67 | 13.70 | 1.701 | 0.1680 | -0.002 |
| G5 | 0.52 | 11.20 | 1.539 | 0.1190 | -0.001 |
| I1 | 0.26 | 18.70 | 2.117 | 0.1420 | 0.001 |
| I2 | 0.24 | 15.30 | 1.891 | 0.1160 | 0.000 |
| I3 | 0.30 | 18.50 | 1.908 | 0.1250 | 0.001 |
| I4 | 0.35 | 15.60 | 1.824 | 0.1040 | 0.000 |
| I5 | 0.19 | 19.40 | 2.222 | 0.1580 | -0.003 |
| S1 | 0.15 | 10.50 | 1.522 | 0.1160 | -0.004 |
| S2 | 0.16 | 8.14 | 1.527 | 0.1063 | -0.002 |
| S3 | 0.27 | 12.50 | 1.555 | 0.0930 | -0.002 |
| S4 | 0.16 | 11.00 | 1.573 | 0.0940 | -0.003 |
| S5 | 0.24 | 10.80 | 1.331 | 0.0850 | -0.003 |
| S6 | 0.30 | 11.40 | 1.415 | 0.0930 | -0.004 |

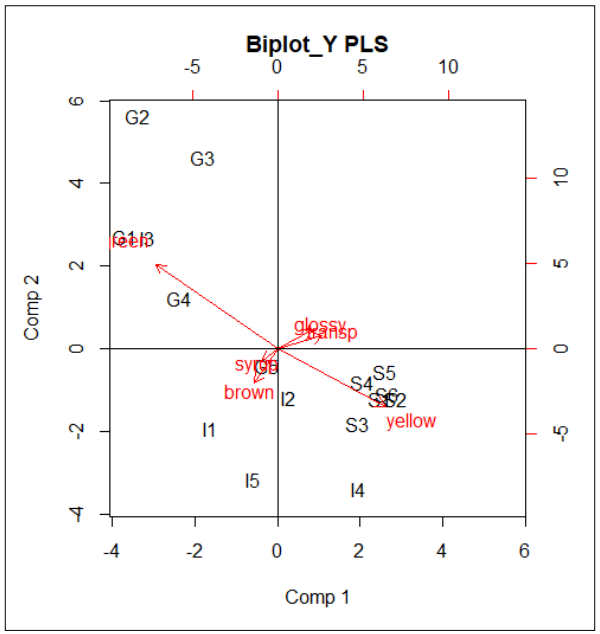
**Grupos** de acordo com a origem do azeite: Grécia (G), Itália (I), Espanha (S)

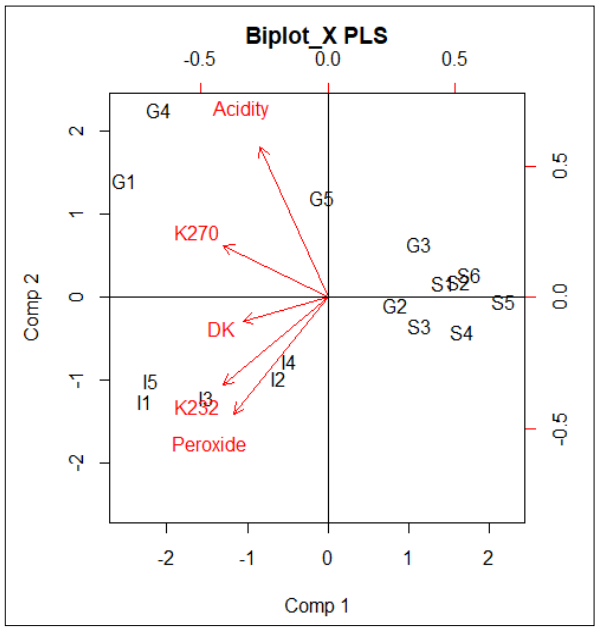
Predição de grupos não é o objetivo da PLS!

$[Y_{16\times6} \quad X_{16\times5}]$

## PLSR

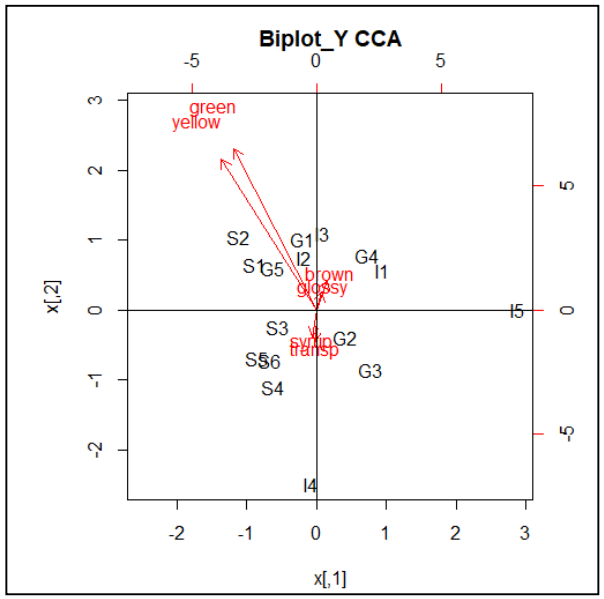
Escores PLS discriminam os grupos (apesar de não ter esse objetivo)

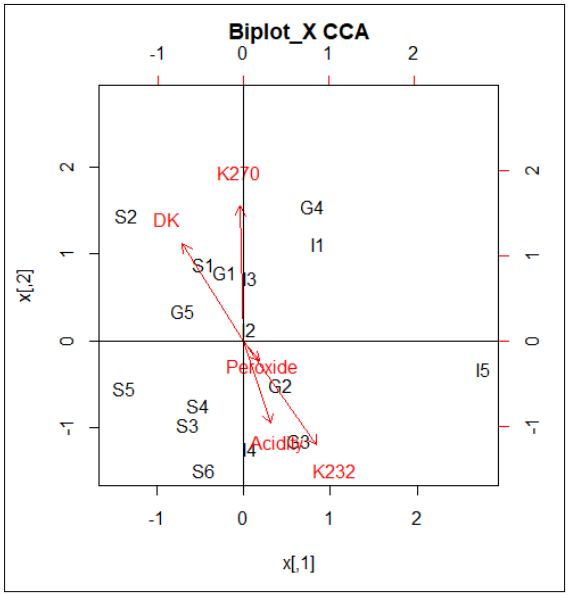


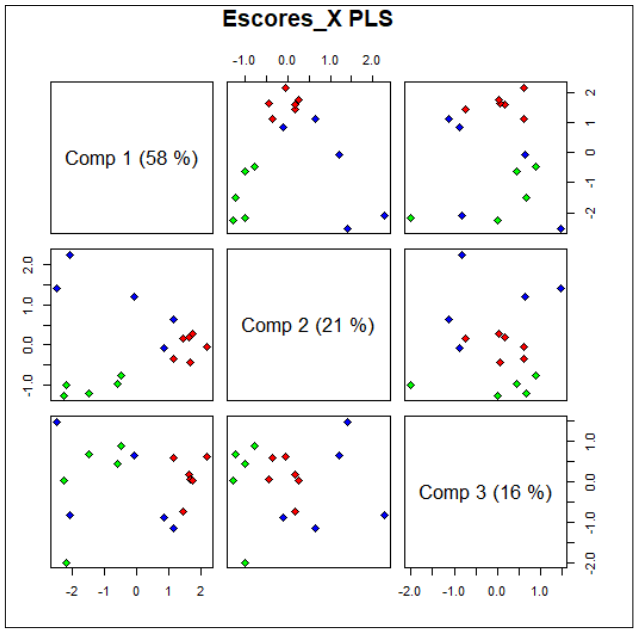

## CCA

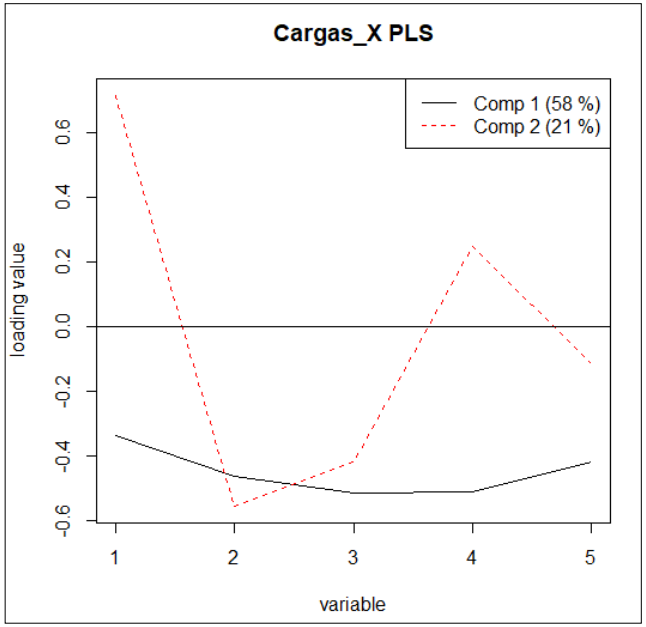
```
r_canônico:
0.98
0.84
0.82
0.57
0.29
```



# PLSR - Oliveoil



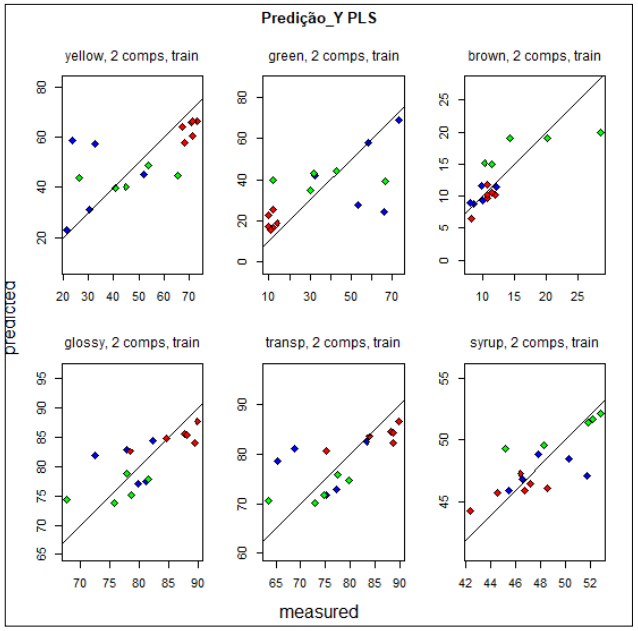



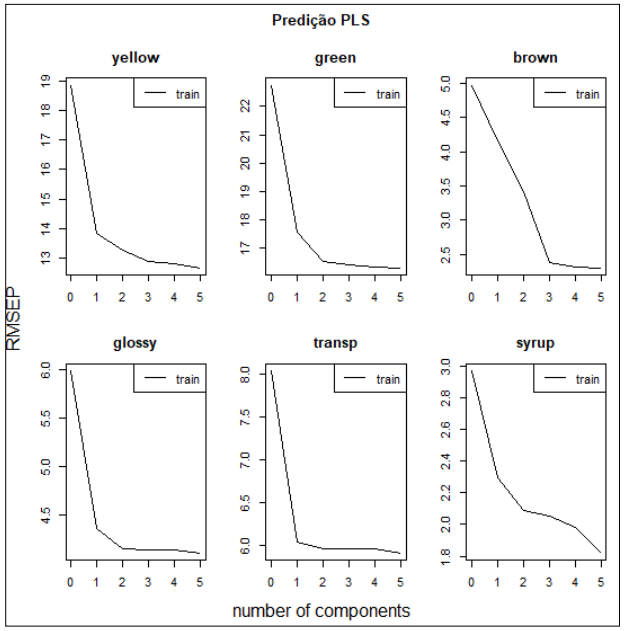

**PLSR**

Neste caso, os escores PLS discriminam os grupos (G, I, S), apesar de não ter esse objetivo.

# PLSR - Oliveoil $[Y_{16\times6} \quad X_{16\times5}]$

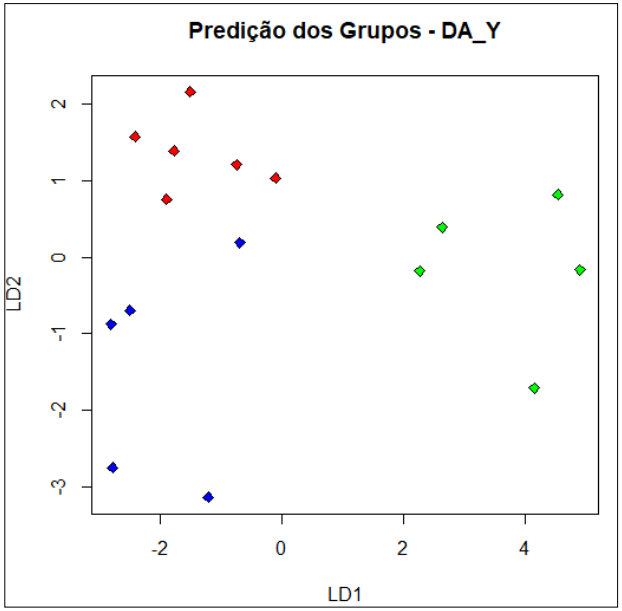
**PLSR**

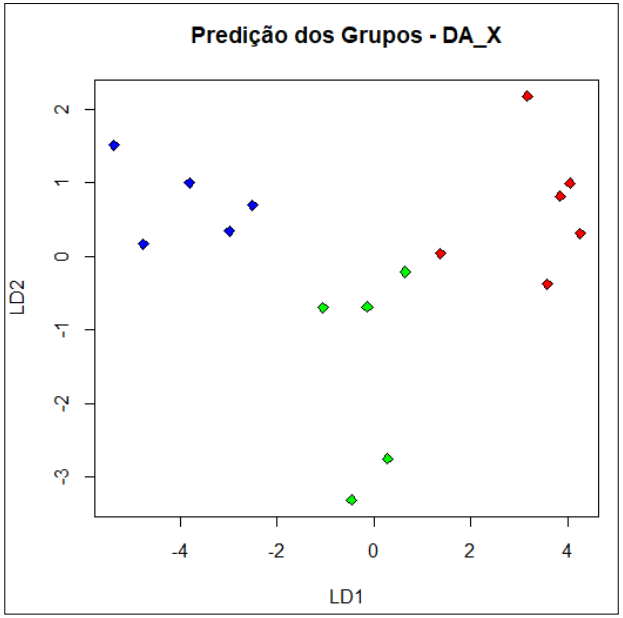



Explorar os recursos do `pls` do R

**AD**




| gr | 1 | 2 | 3 |
|---|---|---|---|
| 1 | 4 | 0 | 1 |
| 2 | 0 | 5 | 0 |
| 3 | 0 | 0 | 6 |

%ClCorreta=94

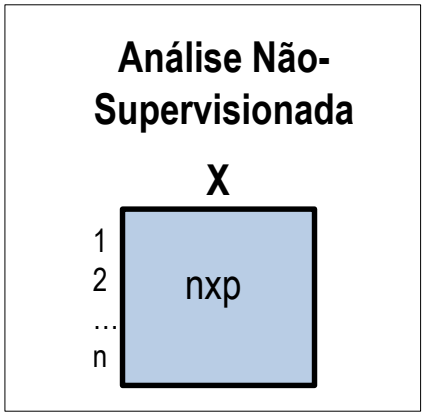
| gr | 1 | 2 | 3 |
|---|---|---|---|
| 1 | 5 | 0 | 0 |
| 2 | 0 | 5 | 0 |
| 3 | 0 | 0 | 6 |

%ClCorr=1

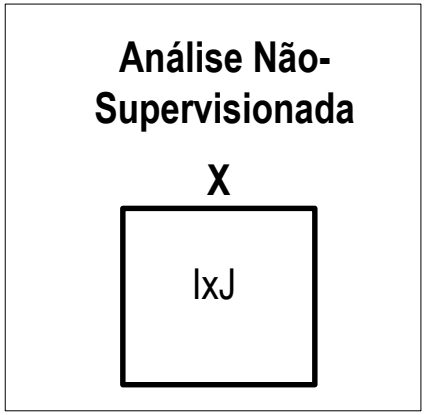
# Análises Multivariadas: Redução de Dimensionalidade e Integração de Bancos de Dados

Dados Quantitativos

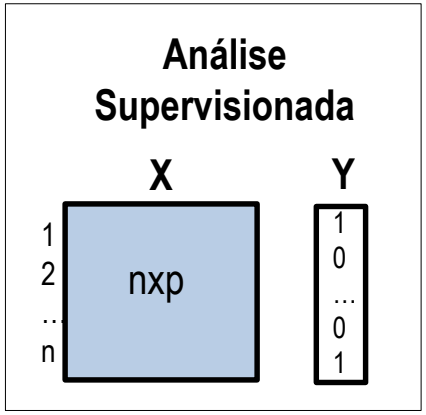
Dados Categóricos



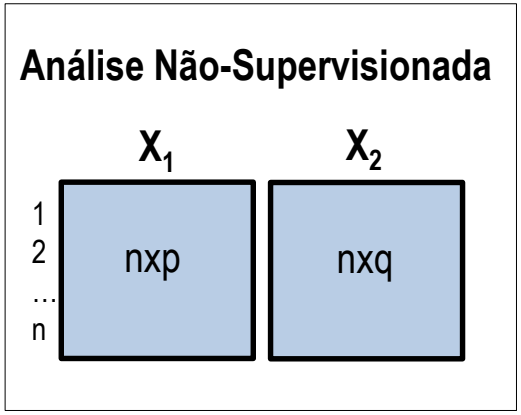

CP, CoP, AF



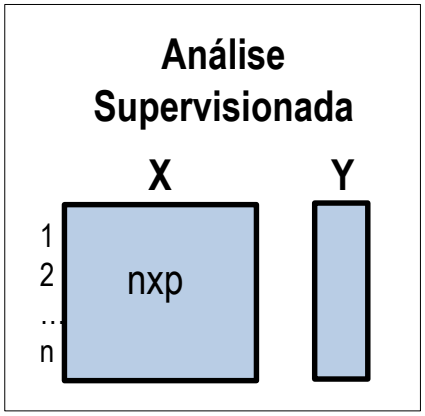

AC



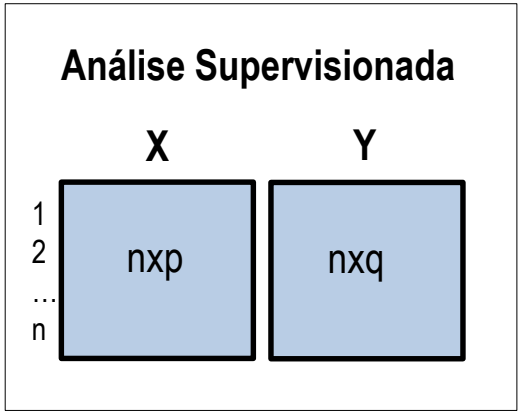

AD



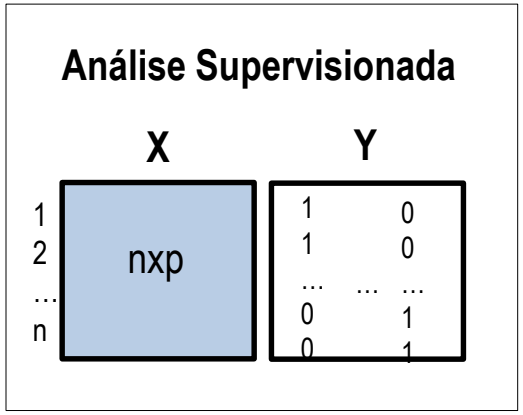

ACC




PLS




PLS




ACC_AD

# O Problema n << p

Big data
Big-p

- **Dados "gasoline"**: $[Y_{60\times1} \quad X_{60\times401}]$

- **Dados dos 3 Experts**:

$$[Y1_{6\times3} \quad Y2_{6\times4} \quad Y3_{6\times3}] = Y_{6\times10}$$

TABLE 1.
Wine tasting data from Abdi and Valentin (2007).

| | | Expert 1 | | | Expert 2 | | | | Expert 3 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Wine | Oak-type | Fruity | Woody | Coffee | Red fruit | Roasted | Vanillin | Woody | Fruity | Butter | Woody |
| 1 | 1 | 1 | 6 | 7 | 2 | 5 | 7 | 6 | 3 | 6 | 7 |
| 2 | 2 | 5 | 3 | 2 | 4 | 4 | 4 | 2 | 4 | 4 | 3 |
| 3 | 2 | 6 | 1 | 1 | 5 | 2 | 1 | 1 | 7 | 1 | 1 |
| 4 | 2 | 7 | 1 | 2 | 7 | 2 | 1 | 2 | 2 | 2 | 2 |
| 5 | 1 | 2 | 5 | 4 | 3 | 5 | 6 | 5 | 2 | 6 | 6 |
| 6 | 1 | 3 | 4 | 4 | 3 | 5 | 4 | 5 | 1 | 7 | 5 |

- **Dados dos Ratos Congenitos**: $Y_{50\times35.129}$

- **Dados dos Transcriptomas**: $Y_{189\times22.215}$

```
library(devtools)
install_github("genomicsclass/tissuesGeneExpression")
library(tissuesGeneExpression)
data(tissuesGeneExpression)
dim(e)   ## e contains the expression data
## [1] 22215   189
```

Lembra disso?

# Componentes Principais – n << p

Big data
Big-p

$$Y_{n\times p} = U_{n\times n}\begin{pmatrix}\Lambda_m^{1/2} & 0\\ 0 & 0\end{pmatrix}V'_{p\times p}$$

Equivalência $\Leftrightarrow$

CP CoP

$$Y_{n\times p}V_{p\times m} = U_{n\times n}\Lambda_m^{1/2}$$

$m \leq min(n,p)$ $\quad Y_{n\times p} \approx U_{n\times m}\Lambda_m^{1/2}V'_{m\times p}$

Os Componentes Principais podem ser obtidos da análise em $\Re^{pxp}$ ou $\Re^{nxn}$ .

Para n > p: realizar a análise em $\Re^{pxp}$ (Decomposição de S: PCA clássico)

Para n < p: realizar a análise em $\Re^{nxn}$ (Decomposição da matriz B obtida de D: Escalonamento Multidimensional ou Coordenadas Principais)

Para n << p: há interessante em soluções penalizadas para eliminar variáveis redundantes da análise, isto é, obter autovetores V que atribuem carga nula a algumas variáveis.

# Componentes Principais – n << p

Big data
Big-p

É facilmente obtido

$$Y_{n\times p} = U_{n\times n}\begin{pmatrix}\Lambda_m^{1/2} & 0\\ 0 & 0\end{pmatrix}V'_{p\times p} \quad \text{Equivalência} \quad \Leftrightarrow \quad Y_{n\times p}V_{p\times m} = U_{n\times n}\Lambda_m^{1/2}$$

$$Y = U\Lambda^{1/2}V'; \quad Z_j = Y V_j \overset{n<<p}{\Rightarrow} \quad Z_j = U_j d_j^{1/2}$$

j-ésimo Componente Principal de $\Re^{nxn}$.

Considere: $\Lambda=(dj)$

Como obter os autovetores tais que, $V_j \cong \nu_j = \left(\nu_{1j}, \nu_{2j}, ..., \nu_{pj}\right); \nu_{kj} = 0$ para muitas coordenadas k mantendo uma alta porcentagem da variância total explicada ?

Primeiro, é interessante estabelecer a CORESPONDÊNCIA entre CP e Modelos de Regressão (Zou, Hastie and Tibshirani, 2006):

CP Regularizado

$$\hat{\beta} = \arg\min_{\beta} \left\| Z_j - Y\beta \right\|_2^2 + \lambda \left\| \beta \right\|_2^2; \quad \lambda > 0, \; \hat{v}_j = \frac{\hat{\beta}}{\left\| \hat{\beta} \right\|_2}; \quad \hat{v}_j = V_j$$

CP conhecido
(obtido da análise em $\Re^{nxn}$)

Solução regularizada
(não depende de $\lambda$)

$$n > p: \; \lambda = 0; \; \hat{v}_j = V_j$$

# Componentes Principais – n << p

## Correspondência entre CP e Modelos de Regressão (Zou, Hastie and Tibshirani, 2006)

- Componente Principal Regularizado (Ridge Regression)

$$\hat{\beta} = \arg\min_{\beta} \left\| Z_j - Y\beta \right\|_2^2 + \lambda \left\| \beta \right\|_2^2$$

$$\left\| \beta \right\|_2^2 = \sum_{j=1}^{p} \beta_j^2$$

↑ parâmetro de regularização
- $\lambda \to 0$: solução MQ
- $\lambda \to \infty$: $\beta_j \to 0$

$$\hat{v}_j = \frac{\hat{\beta}}{\left\| \hat{\beta} \right\|_2}; \quad \hat{v}_j = V_j$$

Solução NÃO depende de $\lambda$

- Componente Principal Penalizado (LASSO)

Limitação: o número de "pesos" não-nulos é no máximo *n*

$$\hat{\beta} = \arg\min_{\beta} \left\| Z_j - Y\beta \right\|_2^2 + \lambda \left\| \beta \right\|_1$$

$$\hat{v}_j = \frac{\hat{\beta}}{\left\| \hat{\beta} \right\|_2}; \quad \hat{Z}_j = Y\,\hat{v}_j$$

↑ parâmetro de penalização
- $\lambda \to 0$: solução MQ
- $\lambda \to \infty$: $\beta_j \to 0$

$$\left\| \beta \right\|_1 = \sum_{j=1}^{p} \left| \beta_j \right|$$

Distância absoluta (norma $L_1$) do vetor $\beta$ à origem

# Componentes Principais – n << p

CP Penalizado (LASSO) x CP Regularizado (Ridge Regression)

Penalização na forma de Lagrange:

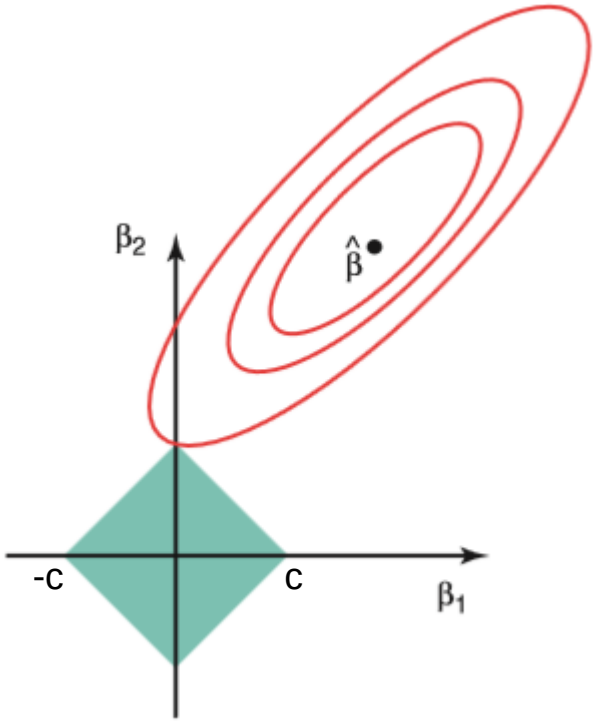
$$\hat{\beta} = \arg\min_{\beta} \left\| Z_j - Y\beta \right\|_2^2 + \lambda \left\| \beta \right\|_1$$

$$\hat{\beta} = \arg\min_{\beta} \left\| Z_j - Y\beta \right\|_2^2 + \lambda \left\| \beta \right\|_2^2$$

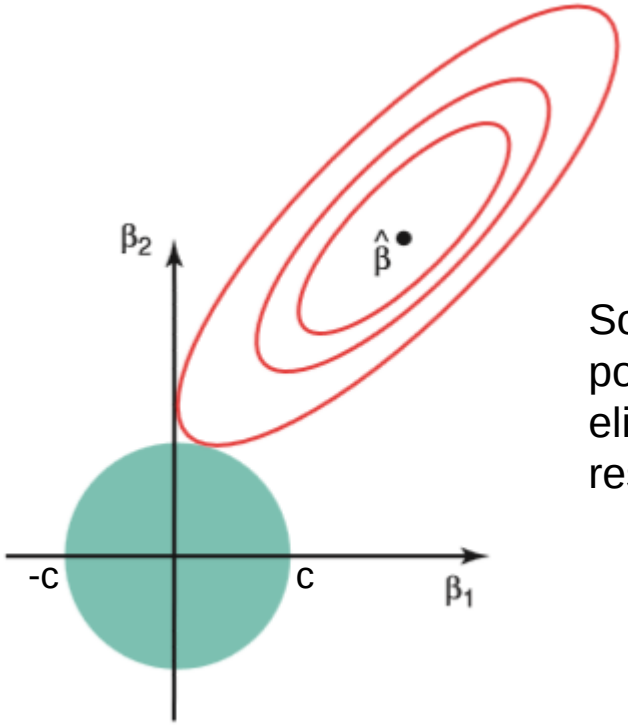
Penalização na forma de restrição:

$$\hat{\beta}_{2\times 1}; \min_{\beta} \sum_{i=1}^{n} \left( Z_{ij} - Y_i' \beta \right)^2 \text{ sujeito a } \left| \beta_1 \right| + \left| \beta_2 \right| \le c$$

$$\hat{\beta}_{2\times 1}; \min_{\beta} \sum_{i=1}^{n} \left( Z_{ij} - Y_i' \beta \right)^2 \text{ sujeito a } \beta_1^2 + \beta_2^2 \le c$$




Solução mais esparsa: $\beta_1 = 0$




Solução menos esparsa: $\beta_1 \approx 0$

Solução: primeiro ponto em que a elipse intercepta a restrição.

# Componentes Principais – n << p

Correspondência entre CP e Modelos de Regressão (Zou, Hastie and Tibshirani, 2006)

Componente Principal Regularizado e Penalizado (Elastic Net):

Vantagem: todas as var podem ser selecionadas (não há limitação no número de "pesos" não-nulos)

$$Y_{n\times p} = U\Lambda^{1/2}V' \overset{n<<p}{\Rightarrow} Z_j = U_j d_j^{1/2} \Rightarrow \hat{Z}_j = Y\hat{v}_j$$

$$\hat{\beta} = \arg\min_{\beta} \left\{ \left\| Z_j - Y\beta \right\|_2^2 + \lambda_1 \left\|\beta\right\|_2^2 + \lambda_2 \left\|\beta\right\|_1 \right\}; \quad \hat{v}_j = \frac{\hat{\beta}}{\left\|\hat{\beta}\right\|_2}; \quad \hat{Z}_j = Y\,\hat{v}_j$$

Solução de mínimos quadrados

parâmetro de regularização

parâmetro de penalização

Quando $n > p$: $\lambda_1 = \lambda_2 = 0; \hat{v}_j = V_j$

$\lambda_1; \lambda_2$ : obtidos por validação-cruzada ou fixados

# Componentes Principais – n<<p
# Correspondência com Modelos de Regressão

## Formalização Geral de Componentes Principais via Modelos de Regressão

$$\arg\min_{\alpha,\beta} \sum_{i=1}^{n} \left\| Y_{i_{p\times 1}} - \alpha_{p\times 1}\beta'_{1\times p}\, Y_i \right\|_2^2 + \lambda \left\|\beta\right\|_2^2; \quad \left\|\alpha\right\|_2^2 = 1 \quad \Rightarrow \hat{\beta} \propto V_1$$

$$\arg\min_{A,B} \sum_{i=1}^{n} \left\| Y_{i_{p\times 1}} - A_{p\times m}B'_{m\times p}\, Y_i \right\|_2^2 + \lambda \sum_{j=1}^{m} \left\|\beta_j\right\|_2^2; \quad \lambda > 0, B = \left(\beta_j\right), A'A = I_m$$

$$\Rightarrow \hat{\beta}_j \propto V_j$$

*m* CP Regularizados e Penalizados podem ser obtidos diretamente de Y:

$$\arg\min_{A,B} \sum_{i=1}^{n} \left\| Y_{i_{p\times 1}} - A_{p\times m}B'_{m\times p}\, Y_i \right\|_2^2 + \lambda_1 \sum_{j=1}^{m} \left\|\beta_j\right\|_2^2 + \sum_{j=1}^{m} \lambda_{2j} \left\|\beta_j\right\|_1;$$

parâmetro de regularização

m parâmetros de penalização

$$A'A = I_m; \; B_{p\times m} = (\beta_1, \ldots, \beta_m); \; \hat{v}_j = \frac{\hat{\beta}_j}{\left\|\hat{\beta}_j\right\|_2}; \quad \hat{Z}_j = Y\, \hat{v}_j; \; j = 1, \ldots, m$$

# Componentes Principais – n<<p
# Correspondência com Modelos de Regressão

*m* CP Regularizados e Penalizados podem ser obtidos diretamente de Y:

$$\arg\min_{A,B} \sum_{i=1}^{n} \left\| Y_{i_{p\times 1}} - A_{p\times m} B'_{m\times p}\, Y_i \right\|_2^2 + \lambda_1 \sum_{j=1}^{m} \left\| \beta_j \right\|_2^2 + \sum_{j=1}^{m} \lambda_{2j} \left\| \beta_j \right\|_1;$$

parâmetro de regularização

m parâmetros de penalização

$$A'A = I_m;\ B_{p\times m} = (\beta_1, ..., \beta_m);\ \hat{v}_j = \frac{\hat{\beta}_j}{\left\| \hat{\beta}_j \right\|_2};\quad \hat{Z}_j = Y\ \hat{v}_j;\ j = 1, ..., m$$

PC Esparso: Variância Explicada (Shen and Huang, 2008)

$$\hat{Z} = \left( \hat{Z}_1, \hat{Z}_2, ..., \hat{Z}_m \right);\quad \hat{Z}_j = Y\hat{v}_j$$

$$\hat{V}_{p\times m} = \left( \hat{v}_1, \hat{v}_2, ..., \hat{v}_m \right) \Rightarrow tr\left( \hat{Y}'\hat{Y} \right);\quad \hat{Y}_{n\times p} = Y_{n\times p} \hat{V} \left( \hat{V}'\hat{V} \right)^{-1} \hat{V}$$

$$\Rightarrow \frac{tr\left( \hat{Y}'\hat{Y} \right)}{tr\left( Y'Y \right)}$$

# Componentes Principais – n<<p

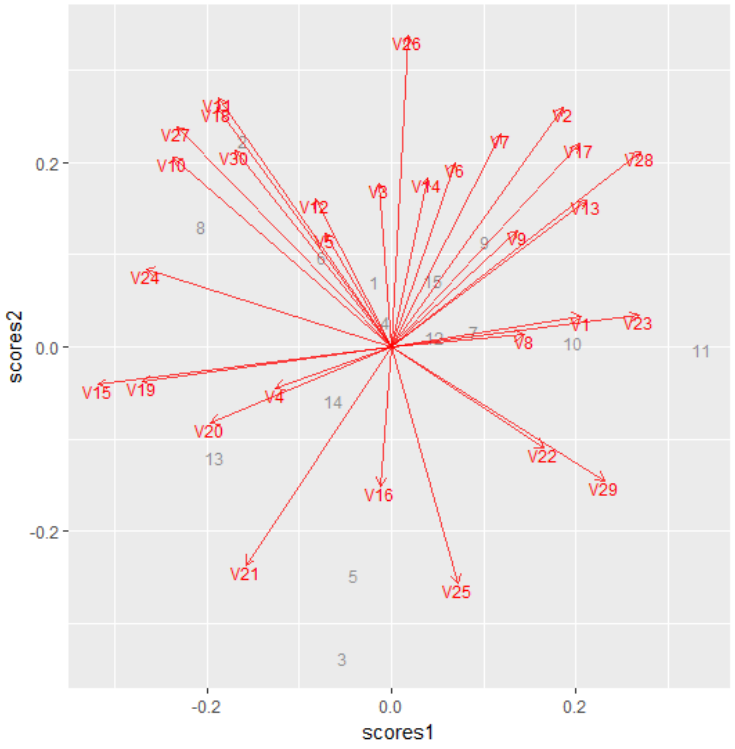
**Sparse Principal Component Analysis**

Hui ZOU, Trevor HASTIE, and Robert TIBSHIRANI



Journal of Computational and Graphical Statistics, Volume 15, Number 2, Pages 265–286
DOI: 10.1198/106186006X113430

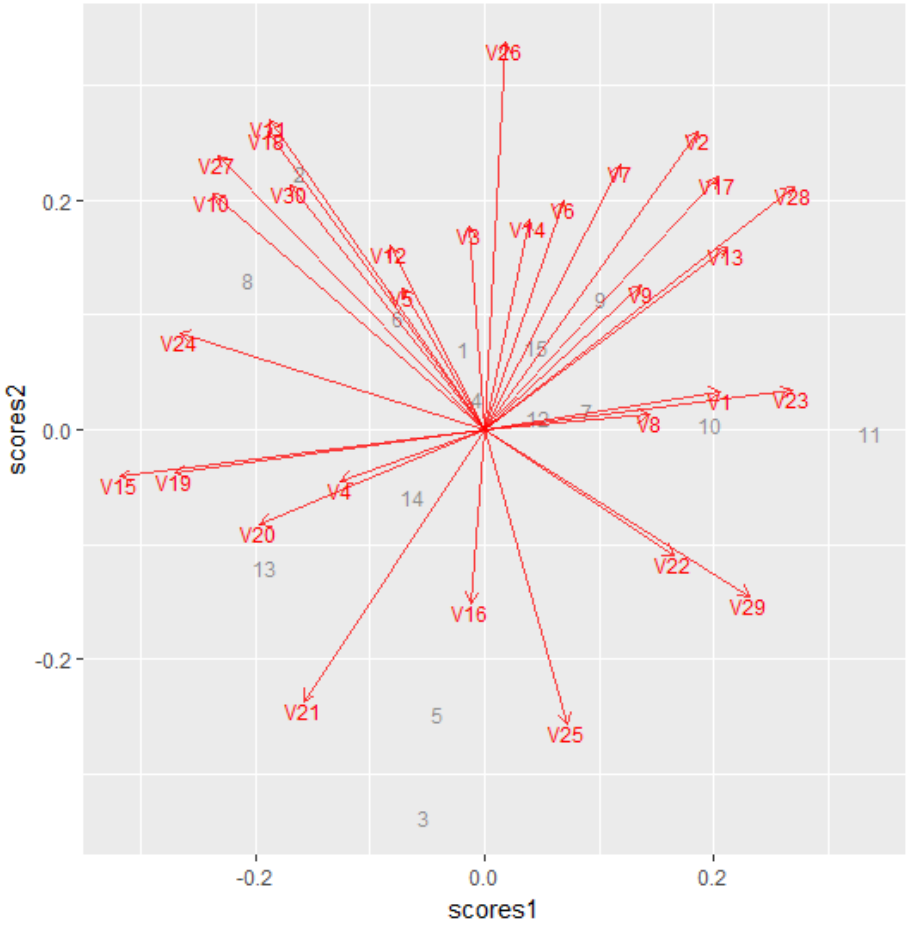
Biplot – (n<p): n=15 p=30
R-prcomp




- *R-SPCA do pacote ElasticNet: Componentes Principais Esparsos*

Solução esparsa: obter autovetores com muitas coordenadas nulas.

# Componentes Principais – n<<p

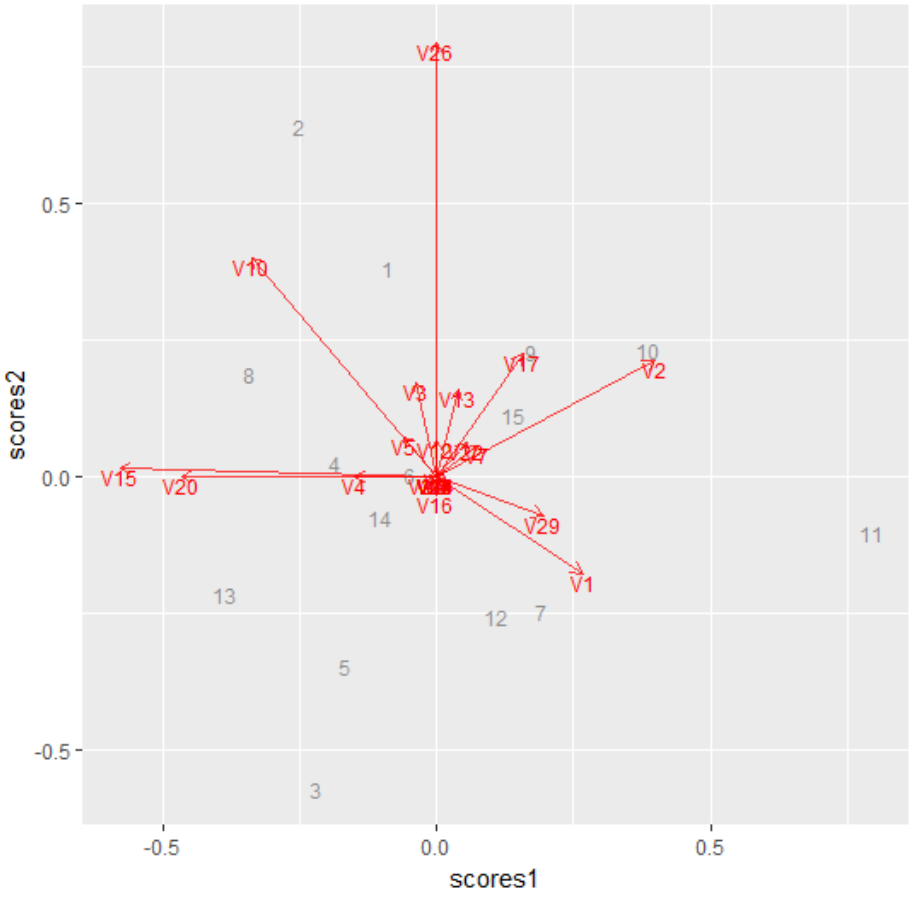



Matriz de cargas

| | PC1 | PC2 |
|---|---|---|
| V1 | 0.20486853 | 0.03338466 |
| V2 | 0.18525221 | 0.26052241 |
| V3 | -0.01406721 | 0.17725332 |
| V4 | -0.12560728 | -0.04474235 |
| V5 | -0.07185638 | 0.12382278 |
| V6 | 0.06894920 | 0.20028957 |
| V7 | 0.11822653 | 0.23051465 |
| V8 | 0.14332703 | 0.01338871 |
| V9 | 0.13705858 | 0.12591629 |
| V10 | -0.23708813 | 0.20630273 |
| V11 | -0.18710753 | 0.26974343 |
| V12 | -0.08342832 | 0.15999298 |
| V13 | 0.21214752 | 0.15918079 |
| V14 | 0.03850244 | 0.18275759 |
| V15 | -0.31794351 | -0.04081232 |
| V16 | -0.01242316 | -0.15170092 |
| V17 | 0.20361151 | 0.22032788 |
| V18 | -0.18979900 | 0.25906266 |
| V19 | -0.26976250 | -0.03717325 |
| V20 | -0.19657886 | -0.08251878 |
| V21 | -0.15850189 | -0.23668719 |
| V22 | 0.16536520 | -0.10948335 |
| V23 | 0.26820473 | 0.03395738 |
| V24 | -0.26656515 | 0.08480063 |
| V25 | 0.07236291 | -0.25674348 |
| V26 | 0.01797063 | 0.33741685 |
| V27 | -0.23208636 | 0.23890523 |
| V28 | 0.26960763 | 0.21224456 |
| V29 | 0.23130349 | -0.14483632 |
| V30 | -0.16921162 | 0.21346408 |

# Componentes Principais – n<<p

Biplot – CP Esparsos: n=15 p=30




Matriz de cargas com esparsidade

```
Sparse loadings
        PC1    PC2
V1    0.266 -0.177
V2    0.398  0.213
V3   -0.040  0.173
V4   -0.151  0.000
V5   -0.062  0.073
V6    0.000  0.000
V7    0.073  0.057
V8    0.000  0.010
V9    0.000  0.000
V10  -0.339  0.401
V11   0.000  0.000
V12   0.000  0.067
V13   0.040  0.160
V14   0.000  0.000
V15  -0.580  0.015
V16   0.000 -0.034
V17   0.157  0.225
V18   0.000  0.000
V19   0.000  0.000
V20  -0.467  0.000
V21   0.000  0.000
V22   0.055  0.063
V23   0.000  0.000
V24   0.000  0.000
V25   0.000  0.000
V26   0.000  0.796
V27  -0.018  0.000
V28   0.000  0.000
V29   0.195 -0.071
V30   0.000  0.000
```

# Componentes Principais – n<<p

Biplot – (n<p): n=15 p=30
R-prcomp