The LAVA Method

Regression for Sparse Models with Dense Perturbations

Julien Chiquet, Rochebrune, mars 2026

Chernozhukov, Victor, Christian Hansen, and Yuan Liao (2017): *A lava attack on the recovery of sums of dense and sparse signals*, The Annals of Statistics

Ćevid, Domagoj, Peter Bühlmann, and Nicolai Meinshausen (2020): *Spectral deconfounding via perturbed sparse linear models.*, Journal of Machine Learning Research

Me and myself, *Lava implementation* in https://github.com/jchiquet/quadrupen R + templated C++

The LAVA Method

Regression for Sparse Models with Dense Perturbations

Julien Chiquet, Rochebrune, mars 2026

- Chernozhukov, Victor, Christian Hansen, and Yuan Liao (2017): *A lava attack on the recovery of sums of dense and sparse signals*, The Annals of Statistics
- Ćevid, Domagoj, Peter Bühlmann, and Nicolai Meinshausen (2020): *Spectral deconfounding via perturbed sparse linear models.*, Journal of Machine Learning Research
- Me and myself, *Lava implementation* in https://github.com/jchiquet/quadrupen R + templated C++

Motivation

Context

- High-dim regression typically relies on **sparsity** assumptions
- Unobserved **confounding variables** makes the coefficients “dense”

💡 Decompose the signal into a **sparse** and a **dense** component

Perturbed Linear Model

The model assumes the response Y is generated by

$$Y = X\theta + \epsilon, \quad \theta = \underbrace{\beta}_{\text{sparse}} + \underbrace{b}_{\text{dense}}, \quad \epsilon \sim \mathcal{N}(0, \sigma^2)$$

- β is a **sparse** vector (the true signal of interest).
- b is a **dense** vector (representing the confounding bias).

⇝ Not identifiable in general (without additional constraints/regularization)

Connexion to Confounding Model and SEMs

Confounding models

High-dim linear systems where unobserved variables H affect both X and Y

$$Y = X\beta + H\delta + \varepsilon$$

where H, X are **jointly Gaussian** with $Cov(H, X) \neq 0$ and ε (sub)-Gaussian

Canonical example: Structural Equation Model (SEM)

By L_2 projection, X can be decomposed as $X = H\Gamma + Z$ where

- $\Gamma = Cov(H)^{-1} Cov(H, X)$, i.e. $H\Gamma = \mathrm{Proj}_H(X)$,
- Z is the unconfounded part of X ($\perp H$, i.e. Cov(H, Z) = 0$).

⇝ We meet back SEM, where β represents the direct causal effect:

$$X \leftarrow H\Gamma + Z, \quad Y \leftarrow X\beta + H\delta + \varepsilon'$$

Mapping to the Perturbed Linear Model

The confounding model can be viewed as a special **perturbed linear model**:

$$Y = X(\beta + b) + (H\delta - Xb) + \varepsilon$$

- The **perturbation** b is defined such that $Xb = \mathrm{Proj}_X(H\delta)$:

$$b = \underbrace{Cov(X)^{-1}}_{Cov(X,H)Cov(H)^{-1}Cov(H,X)+Cov(Z)} Cov(X,H)\delta$$

- The effective error is indeed independen of X:

$$\epsilon = (H\delta - Xb) + \varepsilon = \mathrm{Proj}_X^{\perp}(H\delta) + \varepsilon$$

Interpretation

- Xb is the portion of confounding correlated with X (**bias**)
- $H\delta - Xb$ cannot be explained by X, increasing the error variance

LAVA Loss Function

LAVA estimates $\hat{\theta} = \hat{\beta} + \hat{b}$ by jointly combining ℓ_1 and ℓ_2 penalties:

$$(\hat{\beta}, \hat{b}) = \arg\min_{\beta,b} ||Y - X(\beta + b)||_2^2 + \lambda_1 ||\beta||_1 + \lambda_2 ||b||_2^2$$

Post-LAVA $\tilde{\theta} = \tilde{\beta} + \hat{b}$ removes the bias induced by the ℓ_1-shrinkage:

$$\tilde{\beta} = \arg\min_{\beta} ||Y - X(\beta + \hat{b})||_2^2, \quad \text{such as } \beta_j = 0 \text{ if } \hat{\beta}_j = 0$$

- $\hat{\theta}^{\text{LAVA}} \to b^{\text{Ridge}}$ when $\lambda_1 \to \infty$
- $\hat{\theta}^{\text{LAVA}} \to \beta^{\text{Lasso}}$ when $\lambda_2 \to \infty$

Insight: Orthogonal/univariate Case

Lasso Recall ($\lambda_2 \to \infty$): soft-thresholding

$$\hat{\beta}^{\text{Lasso}} = S(Y, \lambda_1/2), \quad \text{where } S(z, \gamma) = \operatorname{sign}(z) \max(|z| - \gamma, 0)$$

Ridge Recall ($\lambda_1 \to \infty$): simple homothety (shrinkage)

$$\hat{b}^{\text{Ridge}} = \frac{Y}{1 + \lambda_2}$$

LAVA Solution

LAVA first performs (Ridge-style) shrinkage, then Lasso-style thresholding:

$$\hat{\beta}^{\text{LAVA}} = S\left(\frac{Y}{1 + \lambda_2}, \frac{\lambda_1}{2\lambda_2}\right), \hat{b}^{\text{LAVA}} = \frac{Y - \hat{\beta}_{LAVA}}{1 + \lambda_2}, \theta^{\text{LAVA}} = \hat{\beta}^{\text{LAVA}} + \hat{b}^{\text{LAVA}}$$

Univariate Comparison: Lasso vs. LAVA

- **Lasso** must choose: set the coefficient to zero or keeping all the noise/bias
- **LAVA** can capture the “bulk” of the signal via β while “cleaning” the remainder via b.

If Y is very large, $\hat{\beta}$ captures most of it. If Y is small but non-zero, b can absorb it without forcing β to become non-zero.

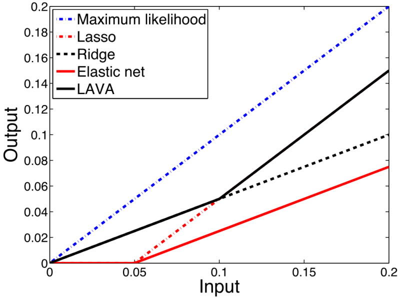


Figure 1: Shrinkage functions implied by various penalized estimators, with $\lambda_1 = \lambda_2 = 1/2$.

Theoretical Risk in Orthogonal case

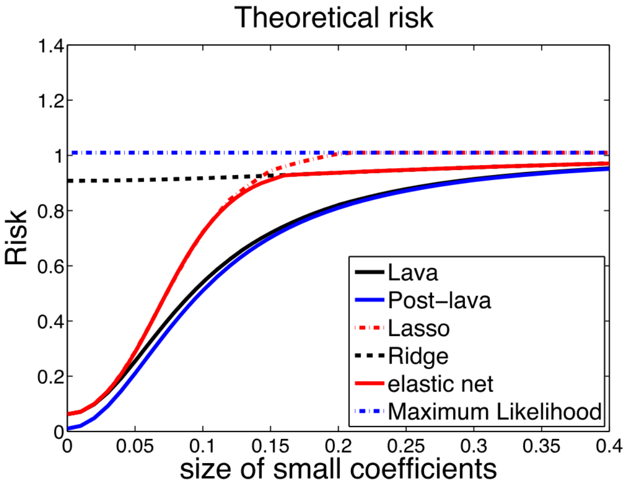


Figure 2: Exact Risk $R(\theta, \hat{\theta}) = \mathbb{E}[\theta - \hat{\theta}]^2$ with oracle choice of the penalty levels for a size-100 $\theta = \beta + b$, $\beta = (3, 0, \dots, 0)$, $b = (0, q, \dots, q)$, as a function of q

LAVA: Calculation Methodology

First, “profile” the loss function wrt b: we get a ridge-type solution on the residuals of β.

$$\hat{b}(\beta) = \arg\min_{b} \|Y - X(\beta + b)\|_2^2 + \lambda_2 \|b\|_2^2 = (X^T X + \lambda_2 I_p)^{-1} X^T (Y - X\beta)$$

Second, consider profiling the loss wrt β:

$$\hat{\beta} = \arg\min_{\beta} \|Y - X(\beta + \hat{b}(\beta))\|_2^2 + \lambda_1 \|\beta\|_1 + \lambda_2 \|\hat{b}(\beta)\|_2^2$$

Theorem 3.1 of LAVA’s paper shows that

$$\hat{\beta} = \arg\min_{\beta} \|\tilde{Y} - \tilde{X}\beta\|_2^2 + \lambda_1 \|\beta\|_1,$$

where

$$\tilde{Y} = P_{\perp}^{1/2}(\lambda_2) Y, \ \tilde{X} = P_{\perp}^{1/2}(\lambda_2) X, \ \text{ with } P_{\perp}(\lambda_2) = I_n - X(X^T X + \lambda_2 \mathbf{S})^{-1} X^T.$$

LAVA: Calculation Methodology

First, "profile" the loss function wrt b: we get a ridge-type solution on the residuals of β.

$$\hat{b}(\beta) = \arg\min_{b} \|Y - X(\beta + b)\|_2^2 + \lambda_2 \|b\|_2^2 = (X^T X + \lambda_2 I_p)^{-1} X^T (Y - X\beta)$$

Second, consider profiling the loss wrt β:

$$\hat{\beta} = \arg\min_{\beta} \|Y - X(\beta + \hat{b}(\beta))\|_2^2 + \lambda_1 \|\beta\|_1 + \lambda_2 \|\hat{b}(\beta)\|_2^2$$

Theorem 3.1 of LAVA's paper shows that

$$\hat{\beta} = \arg\min_{\beta} \|\tilde{Y} - \tilde{X}\beta\|_2^2 + \lambda_1 \|\beta\|_1,$$

where

$$\tilde{Y} = P_{\perp}^{1/2}(\lambda_2) Y, \; \tilde{X} = P_{\perp}^{1/2}(\lambda_2) X, \;\; \text{with } P_{\perp}(\lambda_2) = I_n - X(X^T X + \lambda_2 \mathbf{S})^{-1} X^T.$$

LAVA: Calculation Methodology

First, “profile” the loss function wrt b: we get a ridge-type solution on the residuals of β.

$$\hat{b}(\beta) = \arg\min_{b} \|Y - X(\beta + b)\|_2^2 + \lambda_2 \|b\|_{\mathbf{S}}^2 = (X^T X + \lambda_2 \mathbf{S})^{-1} X^T (Y - X\beta)$$

Second, consider profiling the loss wrt β:

$$\hat{\beta} = \arg\min_{\beta} \|Y - X(\beta + \hat{b}(\beta))\|_2^2 + \lambda_1 \|\beta\|_1 + \lambda_2 \|\hat{b}(\beta)\|_{\mathbf{S}}^2$$

Theorem 3.1 of LAVA’s paper shows that

$$\hat{\beta} = \arg\min_{\beta} \|\tilde{Y} - \tilde{X}\beta\|_2^2 + \lambda_1 \|\beta\|_1,$$

where

$$\tilde{Y} = P_{\perp}^{1/2}(\lambda_2) Y,\ \tilde{X} = P_{\perp}^{1/2}(\lambda_2) X, \text{ with } P_{\perp}(\lambda_2) = I_n - X(X^T X + \lambda_2 \mathbf{S})^{-1} X^T.$$

Generalization: spectral deconfounding

Define $F = K^{1/2}$ a filter and $\tilde{X} = FX, \quad \tilde{Y} = FY$ the transformed data.

With $X = UDV^T$, the singular values of $\tilde{X}$ transformed by LAVA are
$\tilde{d}_i = \frac{\sqrt{\lambda_2} d_i}{\sqrt{\lambda_2 + d_i^2}}$

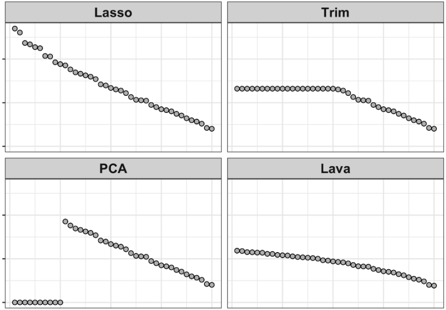


Figure 3: Singular values of $\tilde{X}$ for various spectral transformation/filtering of the data (Cevid et al 20)

Degrees of freedom

Efron degrees of freedom + Stein's SURE theory to measure model's complexity :

$$\mathrm{df} = \frac{1}{\sigma^2} \sum_{i=1}^{n} \mathrm{cov}(y_i, \hat{y}_i) = \mathbb{E}\left[\mathrm{tr}\left(\frac{\partial \hat{y}_i}{\partial y_i}\right)\right]$$

Examples

- $\mathrm{df}(\hat{y}^{\mathrm{ols}}) = \mathrm{tr}(P_X) = p$
- $\mathrm{df}(\hat{y}^{\mathrm{ridge}}) = \mathrm{tr}(P_{\lambda_2})$
- $\mathrm{df}(\hat{y}^{\mathrm{lasso}}) = |\mathcal{A}|$, with $\mathcal{A}$ the set of active (non zero) variables.
- $\mathrm{df}(\hat{y}^{\mathrm{enet}}) = \mathrm{tr}\left(X_{\mathcal{A}}\left(X_{\mathcal{A}}^{\top}X_{\mathcal{A}} + \lambda I_p\right)^{-1} X_{\mathcal{A}}\right)$
- $\mathrm{df}(\hat{y}^{\mathrm{lava}}) = |\mathcal{A}| + \mathrm{tr}\left(\tilde{P}_{\perp}^{\mathcal{A}} P_{\lambda_2}\right)$, with $\tilde{P}^{\mathcal{A}} = \tilde{X}_{\mathcal{A}}\left(\tilde{X}_{\mathcal{A}}^{\top}\tilde{X}_{\mathcal{A}} + \lambda I_p\right)^{-1} \tilde{X}_{\mathcal{A}}$

LAVA: properties

Computational Complexity

Cost of the SVD of X: $O(np\min(n,p))$ + Cost of the LASSO

Theoretical guarantees (Chernozhukov et al. + Cevid et al.)

When $\lambda \asymp \sigma\sqrt{\frac{\log p}{n}}$, under mild conditions, then

$$||\hat{\beta} - \beta||_1 = O_p\left(\frac{\sigma|\mathcal{A}|}{\rho_{\min}(\Sigma)}\sqrt{\frac{\log p}{n}}\right), \quad \text{with } \Sigma = \mathrm{Cov}(X)$$

Simulations with intense confounding

- **Lasso**: quadratic error explodes because it tries to explain the bias b using false variables in β.
- **LAVA**: Maintains low error because the Ridge component absorbs the systematic bias. If no confounding, LAVA is equivalent to Lasso.

Example on multi omics dataset

TCGA Breast Cancer dataset

Public omic data from The Cancer Genome Atlas project

- proteomic data (Reverse Phase protein Array) measuring 163 features
- transcriptomic data (mRNA sequencing) measuring 16749 features

Consider the $n = 174$ patients for which both technologies are available

```r
RPPA_expr %>% as_tibble() %>% head(3) %>% select(1:7) %>% gt() %>% opt_table_font(size="40%")
```

JUN_PS73	YBX1_PS102	YAP1_PS127	SRC_PY527	SRC_PY416	SHC1_PY317	STAT3_PY705
-1.688601	-1.128549	1.2401333	-1.945828	-2.382819	-3.156951	-1.822573
-2.338835	-1.599022	1.8617576	-2.977386	-2.790752	-3.829664	-2.536488
-1.917179	-1.534305	-0.8366558	-2.225239	-2.216812	-3.747675	-2.279488

```r
mRNA_expr %>% as_tibble() %>% head(3) %>% select(1:7) %>% gt() %>% opt_table_font(size="40%")
```

MIR-16/3P	MIR-16/16	MIR-16/5P	MIR-16-1/16	MIR-16-13/16	MIR-16-2/16	GAS8-AS1
0.373875	0.373875	0.373875	0.373875	0.373875	0.373875	0.4308333
0.515750	0.515750	0.515750	0.515750	0.515750	0.515750	0.4460000
-0.177375	-0.177375	-0.177375	-0.177375	-0.177375	-0.177375	-0.1693333

Gene signature of Cancer biomarker

Rationale

- The protein **ERα** plays a key role in regulating cell growth in hormone-sensitive tissues such as the breast
- We look for **transcripts (genes)** that explain variations in ERα
- Omic data are often subject to confounding effect (hidden covariable, bio. process)

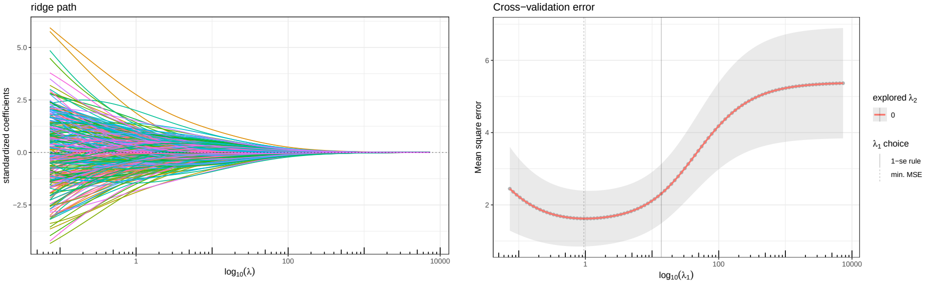
For illustration purpose, we consider a subset of candidates selected by LASSO with small amount of regularization:

```
X <- mRNA_expr; y <- RPPA_expr[, "ESR1"]

screening_out <- quadrupen::lasso(X, y)

screening_out$cross_validate(cores = 1)

X_sub <- X[, which(screening_out$get_model("AIC") != 0)]
```

⇝ We restrict the analysis to the most 264 predictive features among 16749 transcripts

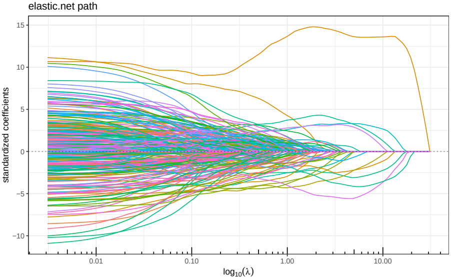
Ridge regression

```
ridge_out <- ridge(X_sub, y)

ridge_out$cross_validate(cores = 1)

model_ridge <- ridge_out$get_model("CV_min")
```

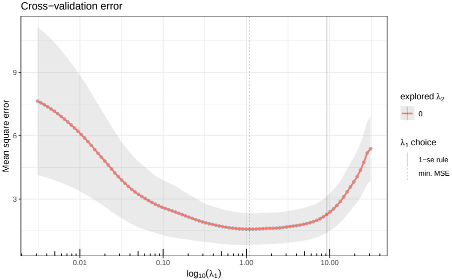


LASSO

```
lasso_out <- lasso(X_sub, y)

lasso_out$cross_validate()

model_lasso <- lasso_out$get_model("CV_min")
```



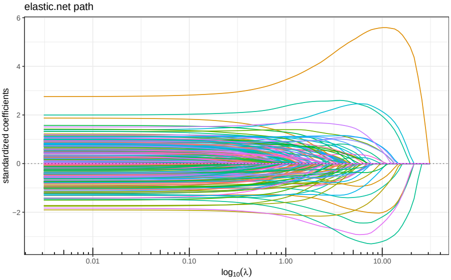
elastic.net path



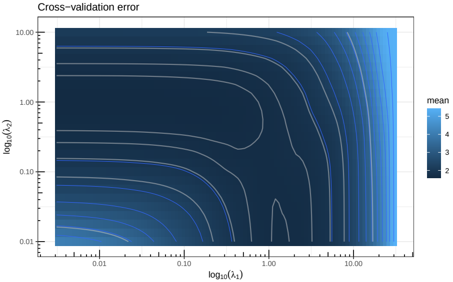
Cross−validation error

Elastic-Net

```
enet_out <- elastic.net(X_sub, y, lambda2=1)

enet_out$cross_validate(lambda2= 10^seq(1,-2,len=25), cores = 1)

enet_final <- elastic.net(X_sub, y, lambda2=enet_out$cross_validation$lambda2_min)

enet_final$cross_validate(cores=1)

model_enet <- enet_final$get_model("CV_min")
```



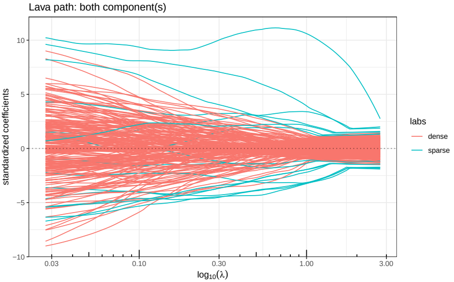
elastic.net path

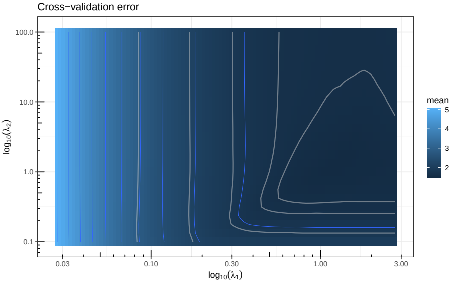


Cross−validation error

Lava

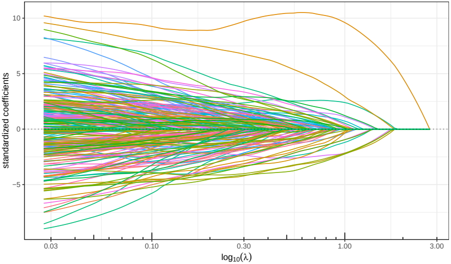
```r
lava_out <- lava(X_sub, y)

lava_out$cross_validate(lambda2 = 10^seq(2,-1,len=25), cores = 1)

lava_final <- lava(X_sub, y, lambda2 = lava_out$cross_validation$lambda2_min)

lava_final$cross_validate(cores=1)

model_lava <- lava_final$get_model("CV_min")
```

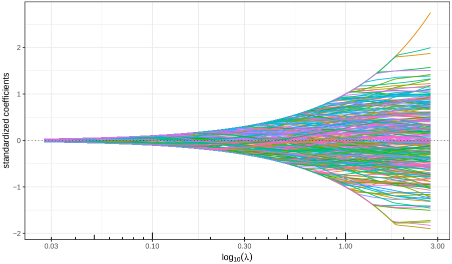




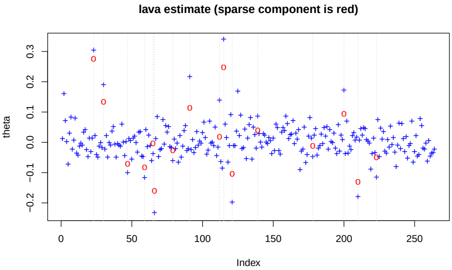
Lava (2)

Closer look at sparse / dense part of the signal

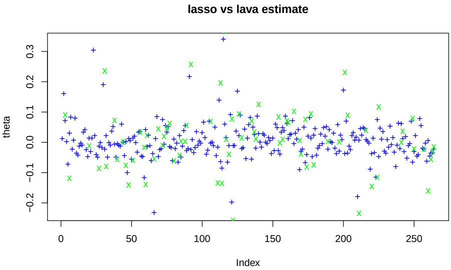

Lava path: sparse component(s)


Lava path: dense component(s)

Lava (3)



lava estimate (sparse component is red)



lasso vs lava estimate

Sparsity and Performance

Number of genes selected by the Elastic-Net: 265

Number of genes selected by the LASSO: 76

Variables selected by Lava: 17

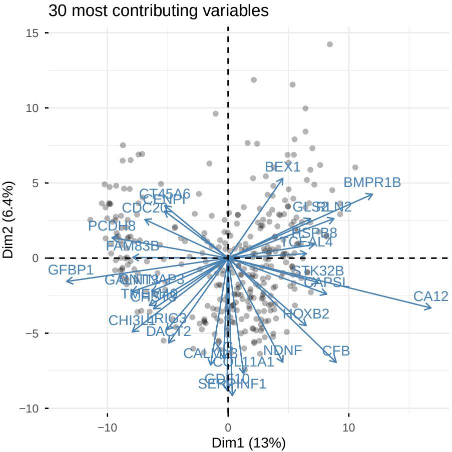
```
 [1] "Intercept" "CA12"      "COL10A1"   "GALNT3"    "KCNK2"     "MGST1"

 [7] "MUC6"      "PKIA"      "HNF1A"     "RNF7"      "DAZAP2"    "CAP1"

[13] "GLS2"      "NDNF"      "ZNF30"     "LRIG3"     "RNF32"
```

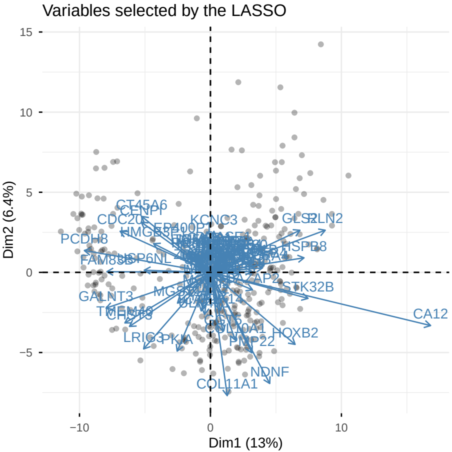
They are 17 selected by Lava that are also selected by the LASSO.

Table 1: Performance

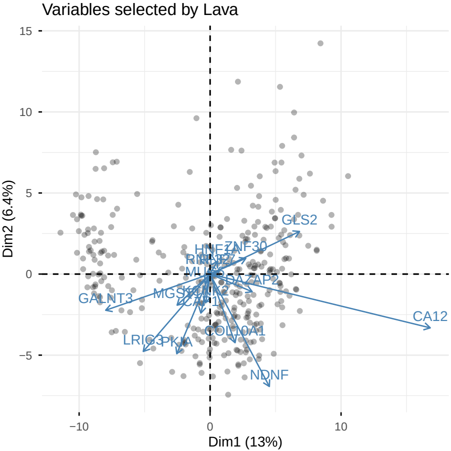
	Ridge	LASSO	Elastic-Net	Lava
MSE	0.4045744	0.4244054	0.4083931	0.4019499
R2	0.8846697	0.8730863	0.8824823	0.8861611
corr	0.9189919	0.9078188	0.9174981	0.9189444
CV_error	1.6212251	1.5723708	1.5269885	1.4774318

Biplot of the design matrix (1)

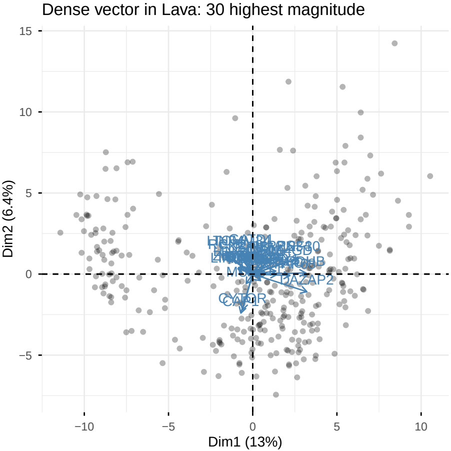

30 most contributing variables


Variables selected by the LASSO

Biplot of the design matrix (2)



Variables selected by Lava



Dense vector in Lava: 30 highest magnitude

Merci de votre attention et aux organisteur/trice ♥♥