Mathematics for Intelligent Systems - 4
21MAT212

End semester Project

Principal Component Pursuit By Alternating Directions

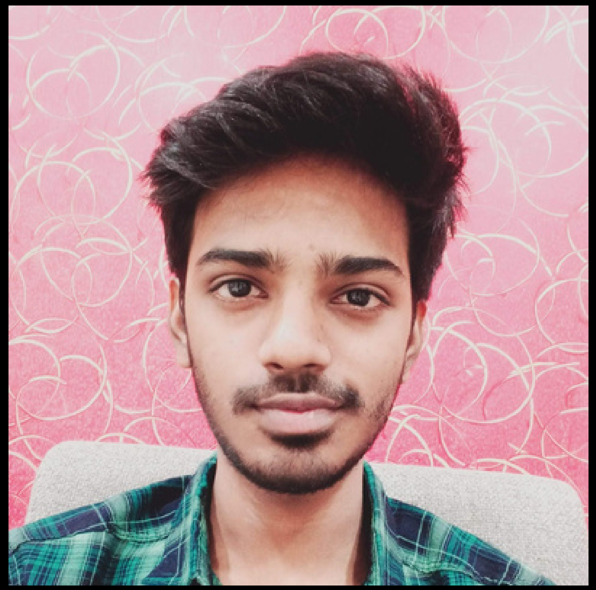
Presented by Team 05

Group Members

SPANDANA

CB.EN.U4AIE20038

NIKHIL



CB.EN.U4AIE20046

PRANAV

CB.EN.U4AIE20053

ROHAN SANJEEV

CB.EN.U4AIE20059

- PRINCIPAL COMPONENT ANALYSIS
- ROBUST PRINCIPAL COMPONENT ANALYSIS
- APPLICATION

PRINCIPAL COMPONENT ANALYSIS

WHAT IS PCA?

Let $X \in \mathbb{R}^{pXn}$ be a data matrix.

Classical PCA looks for the matrix decomposition:

$$\underset{pXn}{X} = \underset{pXn}{L} + \underset{pXn}{E}$$

where

- ***L*** is of low-rank (at most rank ***k***),
- error matrix ***E*** has a small Frobenius norm,
  (which is usually the case for Gaussian noise)

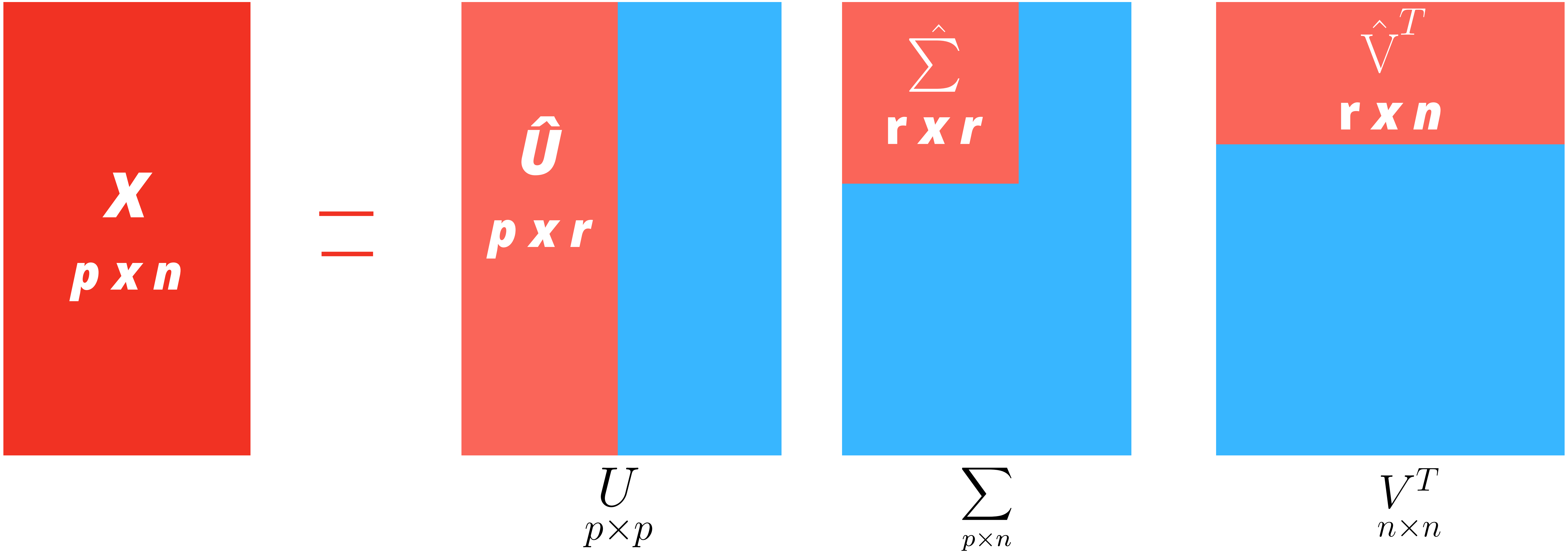
Classical PCA solves

$$\min_{L \in \mathbb{R}^{p \times n}} ||X - L||_2$$

subject to $rank(L) \leqslant k$

Principal Directions

SVD



The rows of V^T or the columns of V are the **principal directions of zero centered X**

$$X = X - mean(X)$$

Principal Directions

$X \in \mathbb{R}^{p \times 2}$ $\quad$ $X^T X$ - correlation matrix

$$X = \begin{bmatrix} | & | \\ x_1 & x_2 \\ | & | \end{bmatrix}, X^T X = \begin{bmatrix} x_1^T x_1 & x_1^T x_2 \\ x_2^T x_1 & x_2^T x_2 \end{bmatrix}$$

$$X = U \Sigma V^T$$

$$X^T X = V \Sigma U^T U \Sigma V^T$$

$$X^T X = V \Sigma^2 V^T$$

$$X^T X V = V \Sigma^2$$

V are the eigen vectors of correlation matrix

EXAMPLE

country	mortality	fertility
Afghanistan	6.82	4.48
Albania	1.33	1.71
Algeria	2.39	2.71
...	...	...
Yemen	5.30	3.89
Zambia	6.16	4.93
Zimbabwe	5.68	3.68

183 rows × 2 columns

$$X \in \mathbb{R}^{183 \times 2}$$

EXAMPLE

country	mortality	fertility
Afghanistan	3.79	1.70
Albania	-1.70	-1.07
Algeria	-0.64	-0.07
...	...	...
Yemen	2.27	1.11
Zambia	3.13	2.15
Zimbabwe	2.65	0.90

183 rows × 2 columns

Mean centered data

$$X = X - mean(X)$$

EXAMPLE

```
U, S, Vt = svd(cntr_child, full_matrices=False)
U, S, Vt
```

```
(array([[-0.1 ,  0.02],
        [ 0.05, -0.04],
        [ 0.01,  0.02],
        ...,
        [-0.06,  0.02],
        [-0.09,  0.1 ],
        [-0.06, -0.02]]),
 array([43.48,  8.2 ]),
 array([[-0.93, -0.37],
        [-0.37,  0.93]]))
```

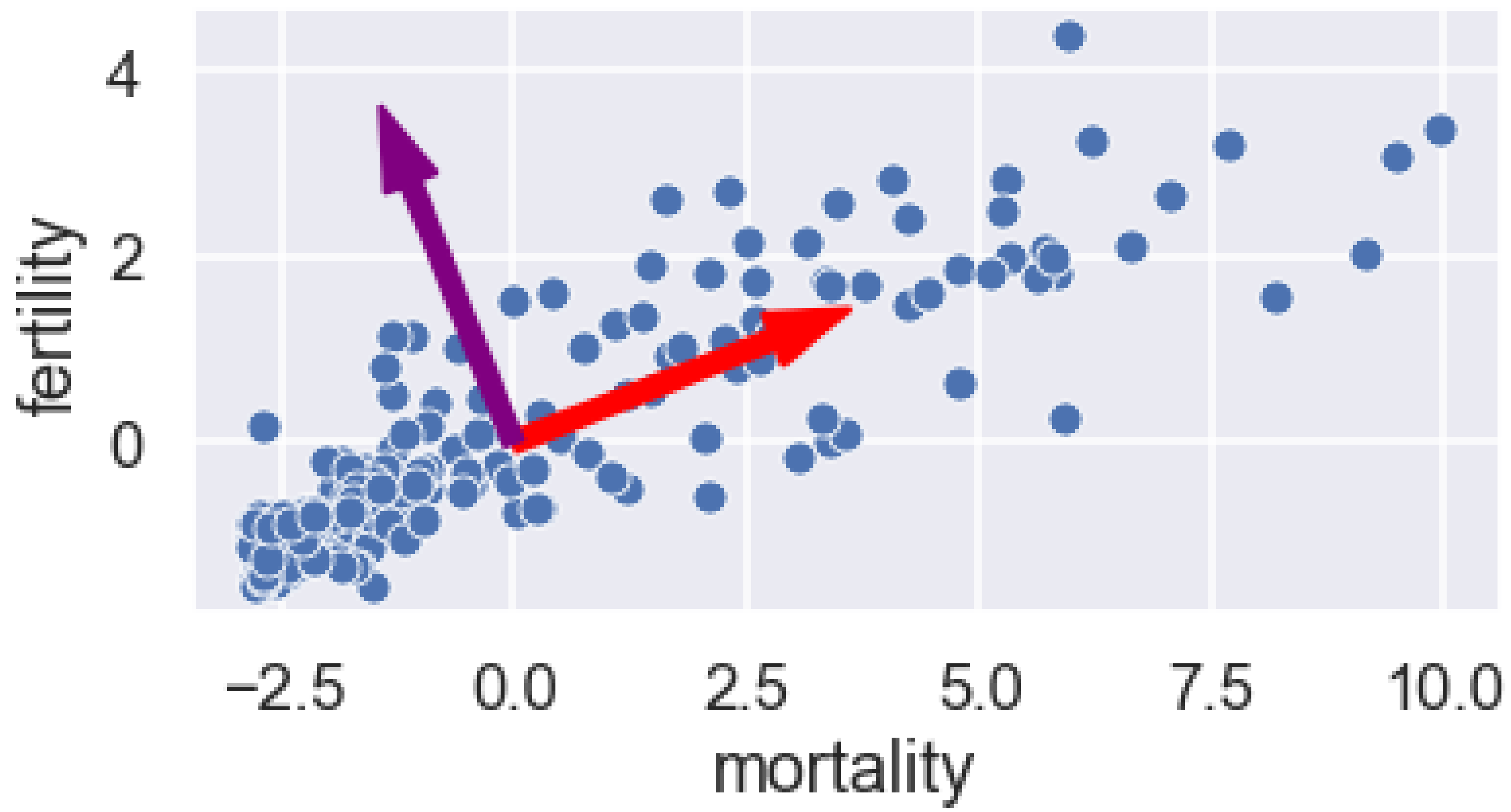
$$U \in \mathbb{R}^{183\times 183}$$
$$\Sigma \in \mathbb{R}^{183\times 2}$$
$$V^T \in \mathbb{R}^{2\times 2}$$

EXAMPLE

$$V^T \in \mathbb{R}^{2\times 2}$$

```
array([[-0.93, -0.37],
       [-0.37,  0.93]])
```

EXAMPLE



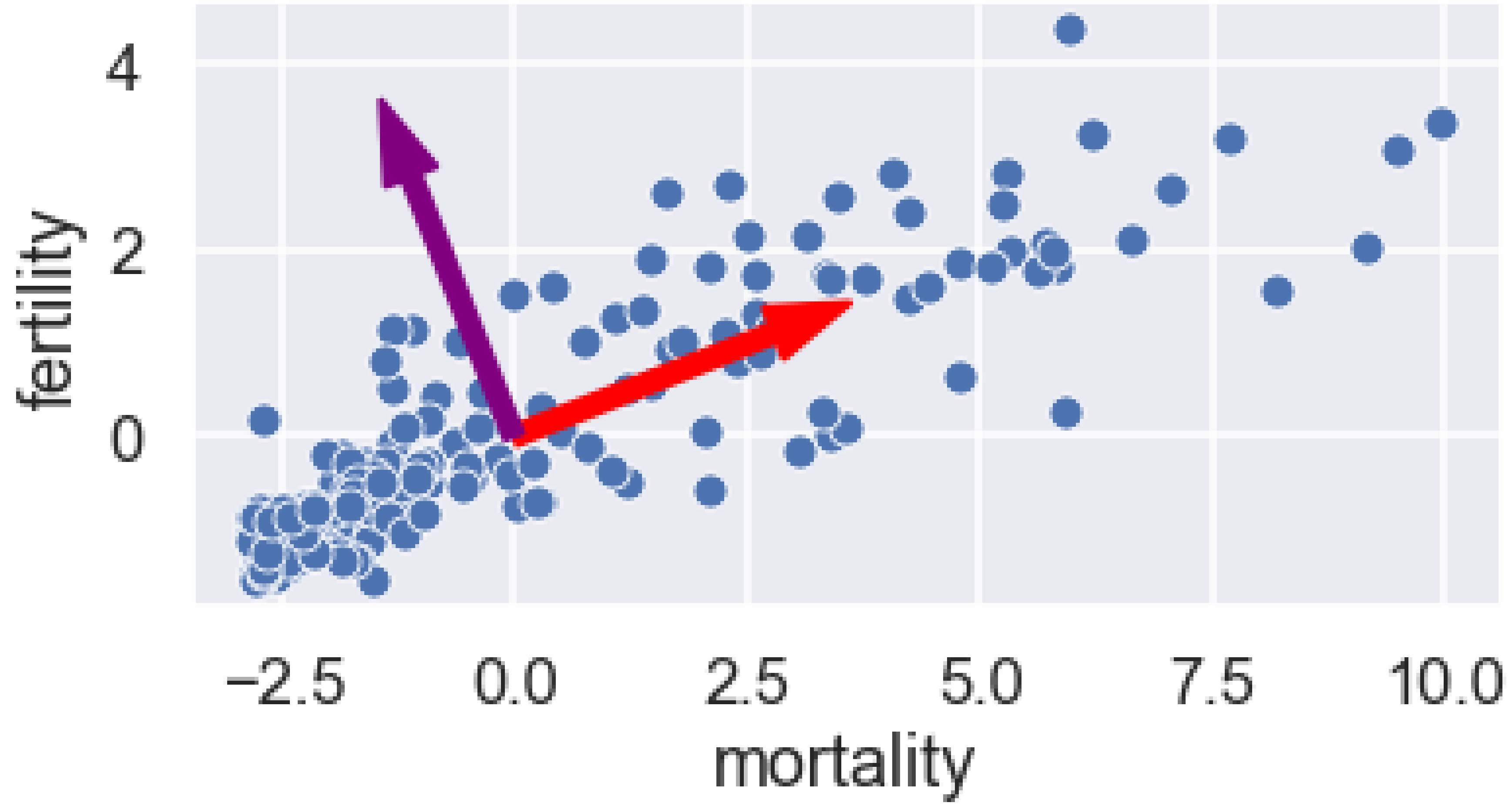

```
array([[-0.93, -0.37],
       [-0.37,  0.93]])
```

Dimensionality reduction

- If **X** has '**n**' dimensions, then **X** will have '**n**' principal directions.
- To reduce the dimension of **X** to '***k***' dimensions, the data is projected onto the first '***k***' principal directions.

This is PCA, a method for dimensionality reduction.

EXAMPLE




```
array([[-0.93, -0.37],
       [-0.37,  0.93]])
```

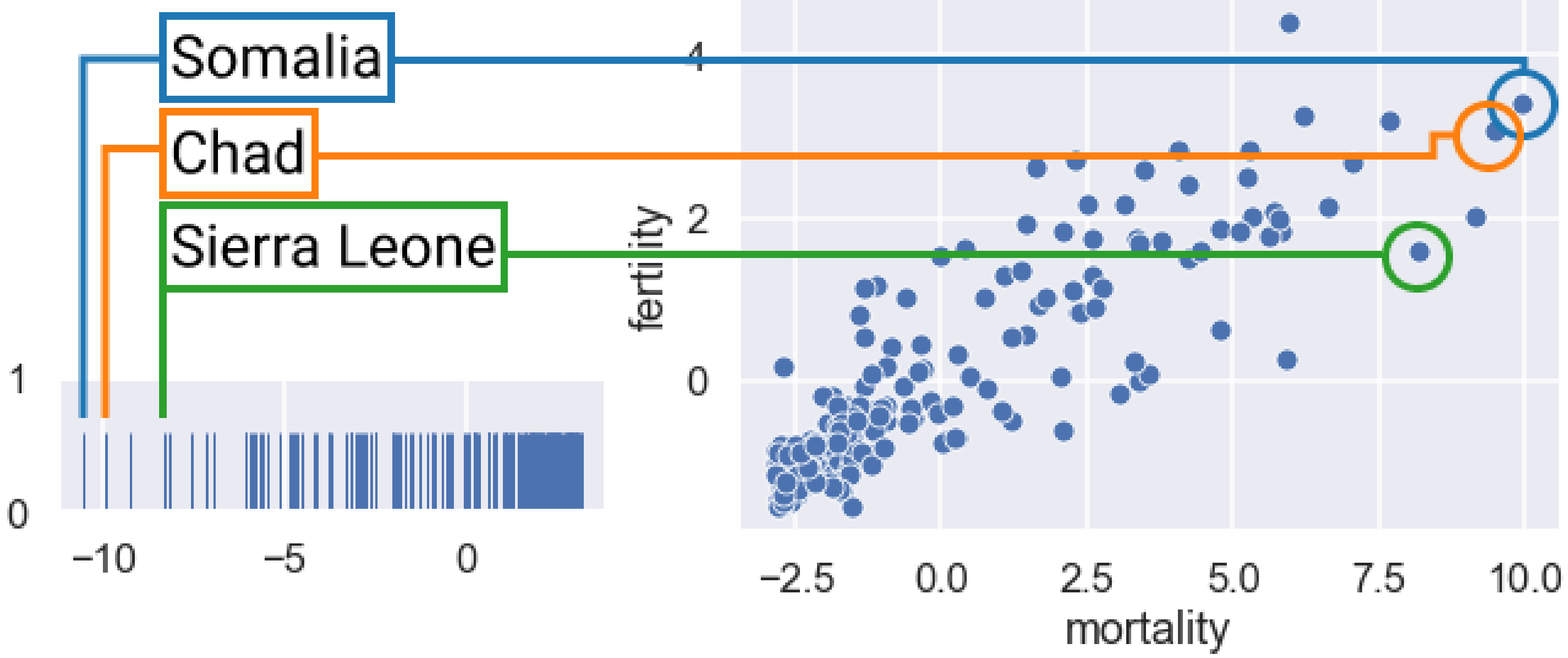
EXAMPLE

$X \in \mathbb{R}^{183\times 2}$ $\quad V_1 \in \mathbb{R}^{2\times 1}$

```
X @ Vt[:1].T
```

```
array([[-4.15],
       [ 1.98],
       [ 0.62],
       ...,
       [-2.52],
       [-3.7 ],
       [-2.79]])
```

$\mathbb{R}^{183\times 1}$



Simpler Approach

Instead of computing XV each time by removing columns of V

$$X = U\Sigma V^T$$
$$XV = U\Sigma V^T V$$
$$XV = U\Sigma$$

Right-multiply by V

Since $V^T V = I$

Keep removing the last column of $U\Sigma$ until desired dimensionality is obtained

Rank "k" approximation

Rank-k Approximation

This method is used to produce a **rank k matrix** with the same dimensions as **X.**

It projects the points in **X** to the **top-k** principal components.

1. Center $\mathbf{X}$.
2. Use the SVD to find $\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top$.
3. Keep the first k columns of $\mathbf{U}\boldsymbol{\Sigma}$, and the first k rows of $\mathbf{V}^\top$.
4. Compute $\mathbf{X} \approx \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top$ with the reduced $\mathbf{U}\boldsymbol{\Sigma}$ and $\mathbf{V}^\top$.

Rank-k Approximation

$$SVD(X) = U\Sigma V^t$$

$$X \in \mathbb{R}^{m\times n}, U \in \mathbb{R}^{m\times m}, \Sigma \in \mathbb{R}^{m\times n}, V^t \in \mathbb{R}^{n\times n}$$

$$U = (U_L|U_R), V = (V_L|V_R), \text{and } \Sigma = \begin{pmatrix} \Sigma_{TL} & 0 \\ 0 & \Sigma_{BR} \end{pmatrix},$$

$$\text{where } U_L \in \mathbb{R}^{m\times k}, V_L \in \mathbb{R}^{n\times k}, \text{and} \Sigma_{TL} \in \mathbb{R}^{k\times k}$$

$$\text{Then } X \approx U_L \Sigma_{TL} V_L^T$$

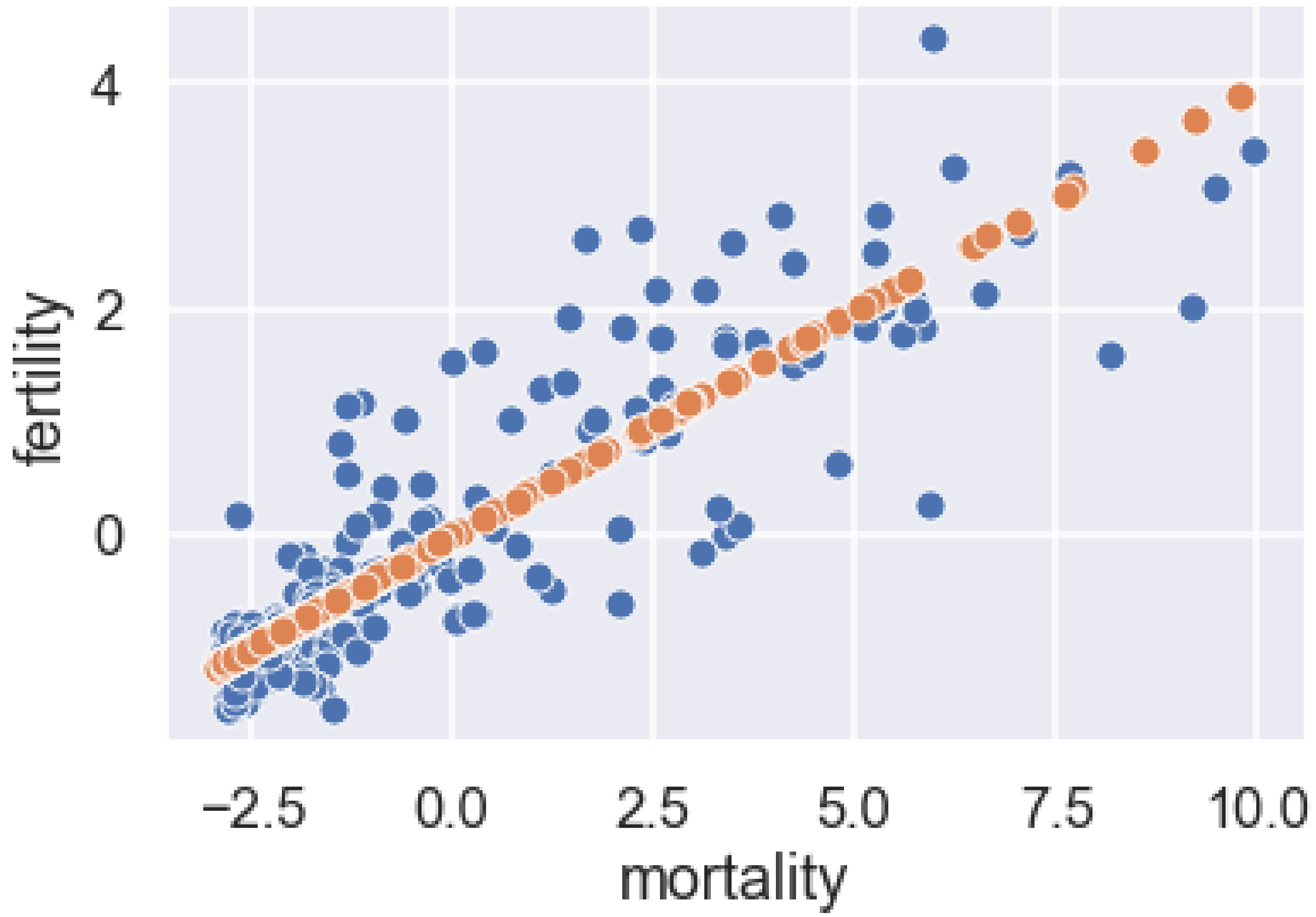
$$m \times n = m \times k * k \times k * k \times n$$

EXAMPLE

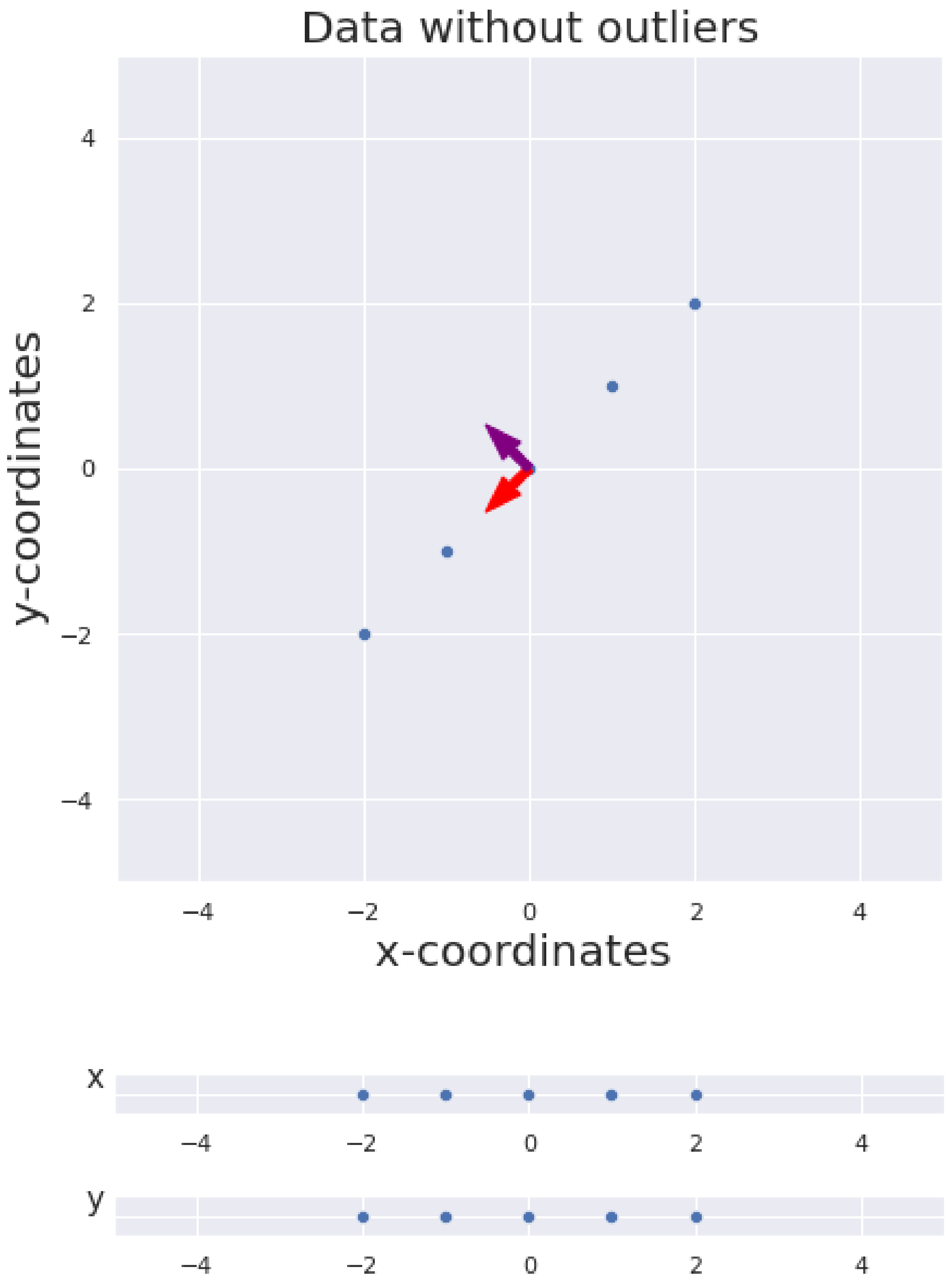
Here K is taken as 1

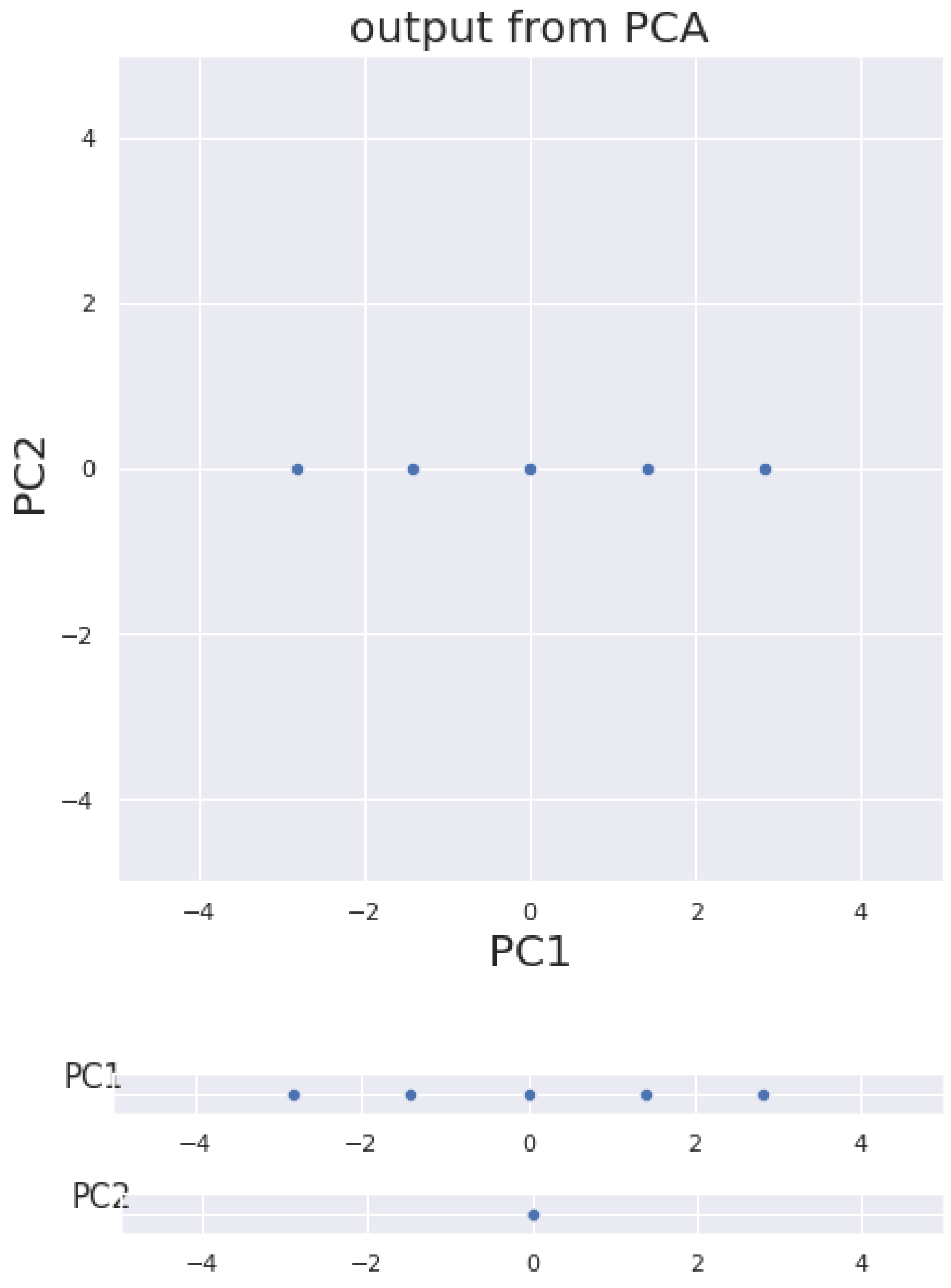
	mortality	fertility
0	3.86	1.53
1	-1.84	-0.73
2	-0.58	-0.23
...	...	...
180	2.34	0.93
181	3.44	1.36
182	2.60	1.03

183 rows × 2 columns

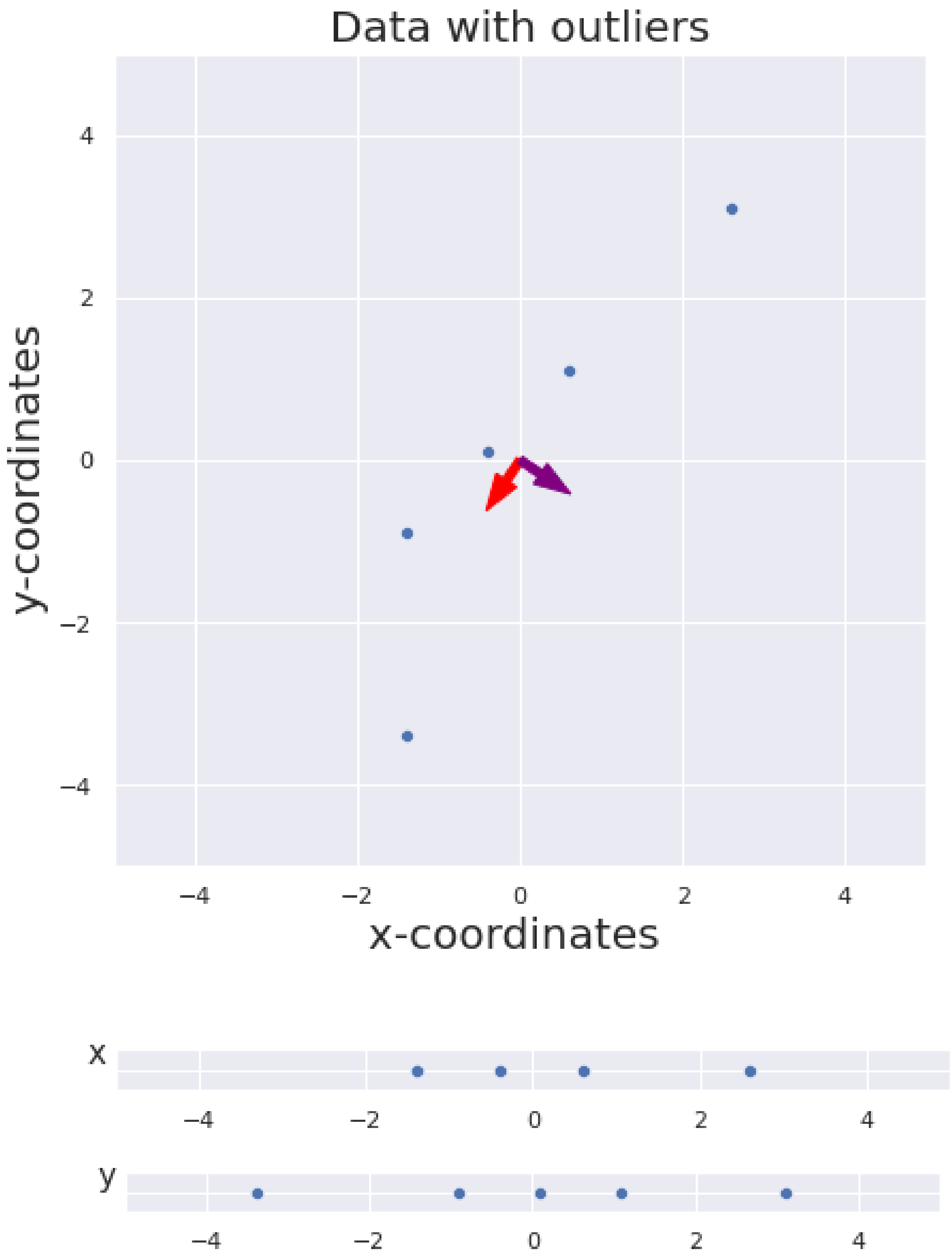


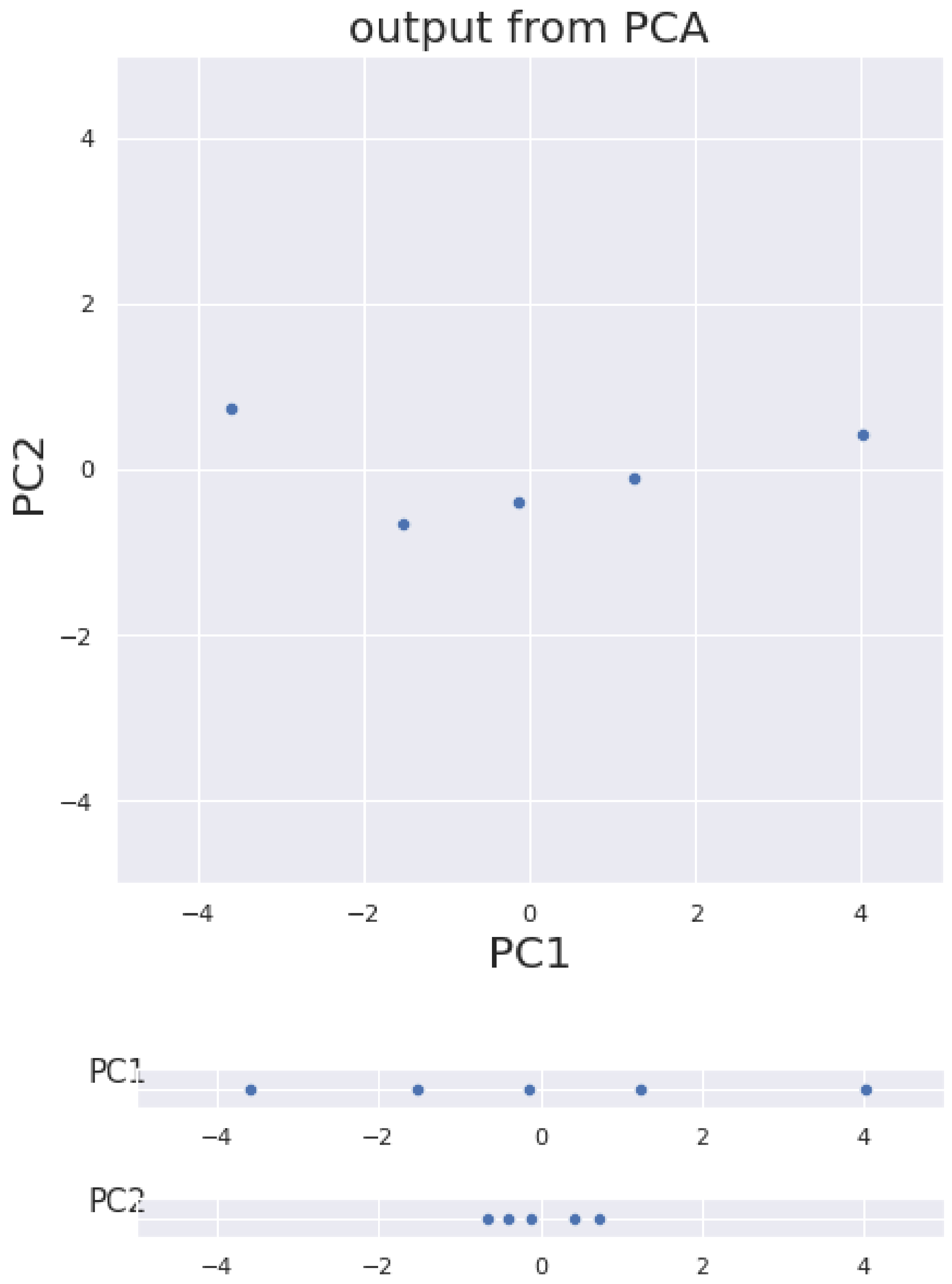
PCA Without Outliers





PCA With Outliers





ROBUST PRINCIPAL COMPONENT ANALYSIS

WHAT IS RPCA?

Let $X \in \mathbb{R}^{pXn}$ be a data matrix. RPCA looks for the following decomposition.

$$\underset{pXn}{X} = \underset{pXn}{L} + \underset{pXn}{S}$$

where

- ***L*** is of low-rank (at most rank ***k***),
- **S** is a sparse matrix

RPCA solves

$$\min_{L,S} \; rank(L) + \|S\|_0$$

$$\text{subject to } L + S = X$$

$$\text{where } \|S\|_0 := \#\{S_{ij} \neq 0\}$$

Convex test

We say that f is convex on S if

$$f(\lambda x + (1-\lambda)y) \;\leq\; \lambda f(x) + (1-\lambda) f(y)$$

$$for\ all\ x, y \in S\ \ and\ all\ \lambda \in (0,1)$$

Convexity of rank()

$$X = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \qquad Y = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \qquad \lambda = 0.5$$

$$rank(\begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix}) \nleq 0.5 * rank(\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}) + 0.5 * rank(\begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix})$$

2   0.5*1 +0.5*1

rank() is non-convex

Convexity of L0 norm

$$\text{Let } x := \begin{bmatrix} 1 \\ 0 \end{bmatrix} \text{ and } y := \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$\text{for any } \lambda \in (0, 1)$$

$$\|\lambda x + (1-\lambda) y\|_0 = 2 \not\leq \lambda \|x\|_0 + (1-\lambda)\|y\|_0 = 1$$

L0 norm is non-convex

Convex relaxation

$$\mathbf{rank}(L) \ := \# \{\sigma_i(L) \neq 0\} \implies \|L\|_* = \sum_i \sigma_i(L),$$

$$\|S\|_0 := \# \{S_{ij} \neq 0\} \implies \|S\|_1$$

Convexity of L1 norm

$$\text{Let } x := \begin{bmatrix} 1 \\ 0 \end{bmatrix} \text{ and } y := \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$\text{for any } \lambda \in (0, 1)$$

$$\|\lambda x + (1 - \lambda) y\|_1 = 1 \leq \lambda \|x\|_1 + (1 - \lambda) \|y\|_1 = 1$$

L1 norm is convex

Convexity of nuclear norm

$$\text{Let } x := \begin{bmatrix} 1 \\ 0 \end{bmatrix} \text{ and } y := \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$\text{for any } \lambda \in (0, 1)$$

$$||\lambda x + (1-\lambda)y)||_* \leq \lambda\, ||x||_* + (1-\lambda)\, ||y||_*$$

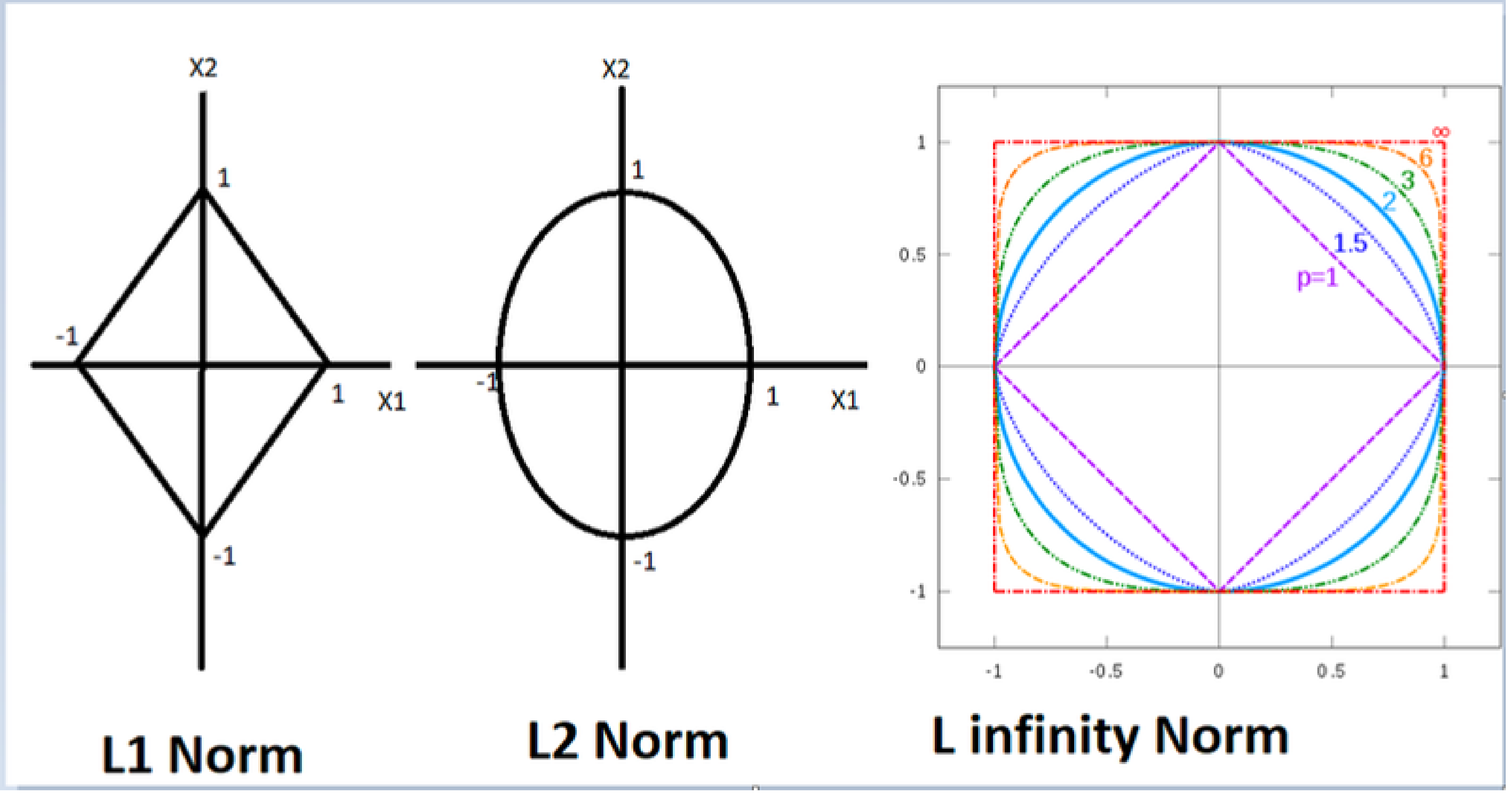
$\lambda = 0.1$	$\lambda = 0.5$	$\lambda = 0.9$
$\begin{bmatrix} 0.9055 \\ 0 \end{bmatrix} \leq \begin{bmatrix} 1 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 0.7071 \\ 0 \end{bmatrix} \leq \begin{bmatrix} 1 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 0.9055 \\ 0 \end{bmatrix} \leq \begin{bmatrix} 1 \\ 0 \end{bmatrix}$

Nuclear norm is convex

Inducing Sparsity using L1 Norm

L1 norm is defined as the summation of absolute values of a vector's all components.

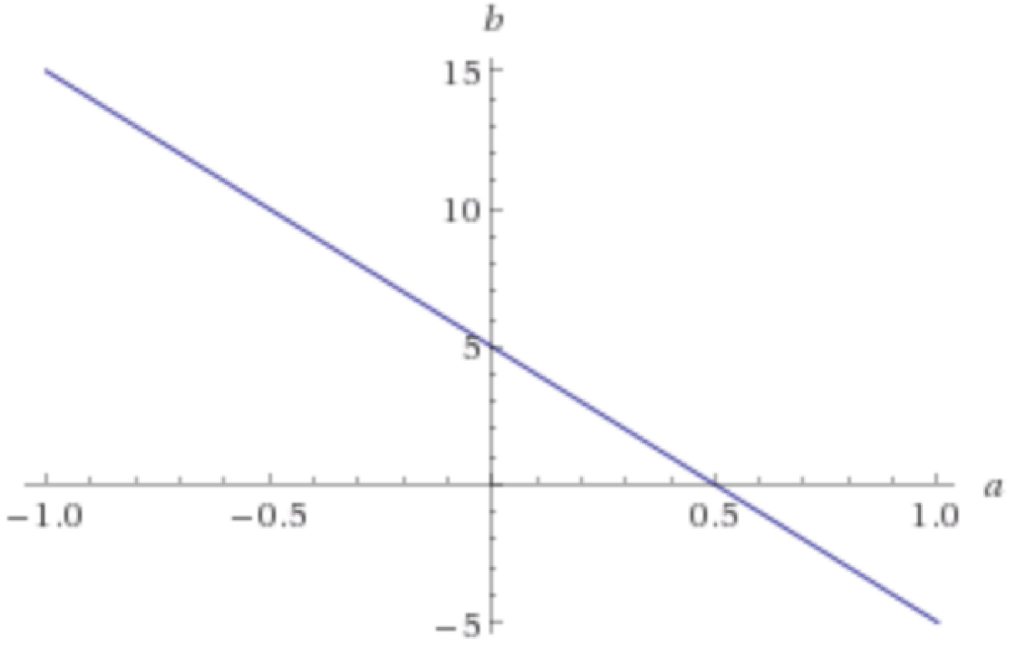
If a vector is [x, y], its L1 norm is |x| + |y|



L1 Norm has spikes that happen to be at sparse points..

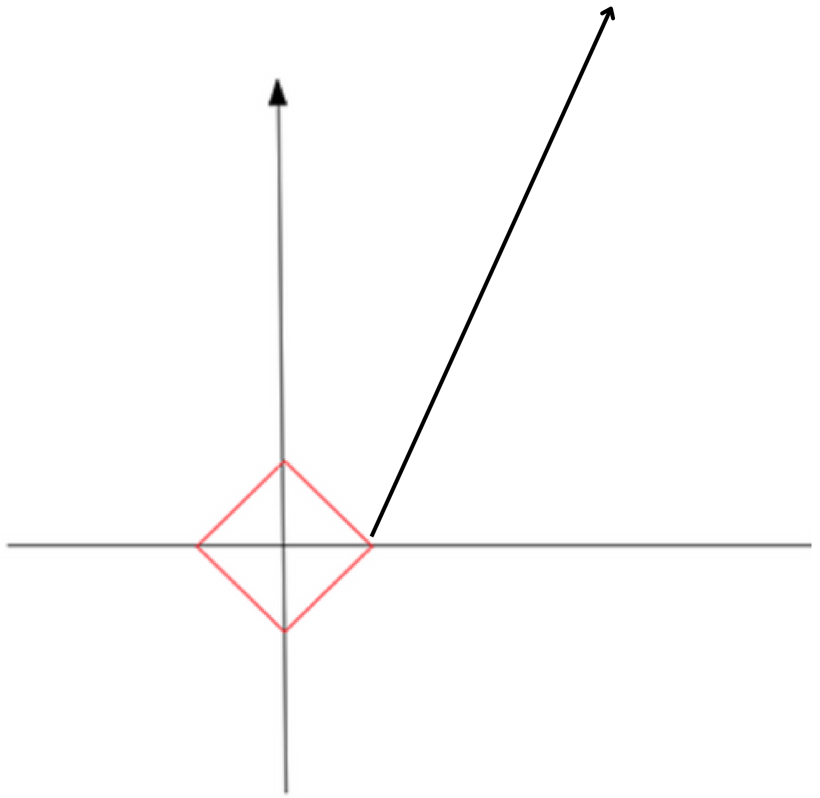
Inducing Sparsity using L1 Norm

$$[10 \quad 1] \times \begin{pmatrix} a \\ b \end{pmatrix} = 5$$



10a + b = 5

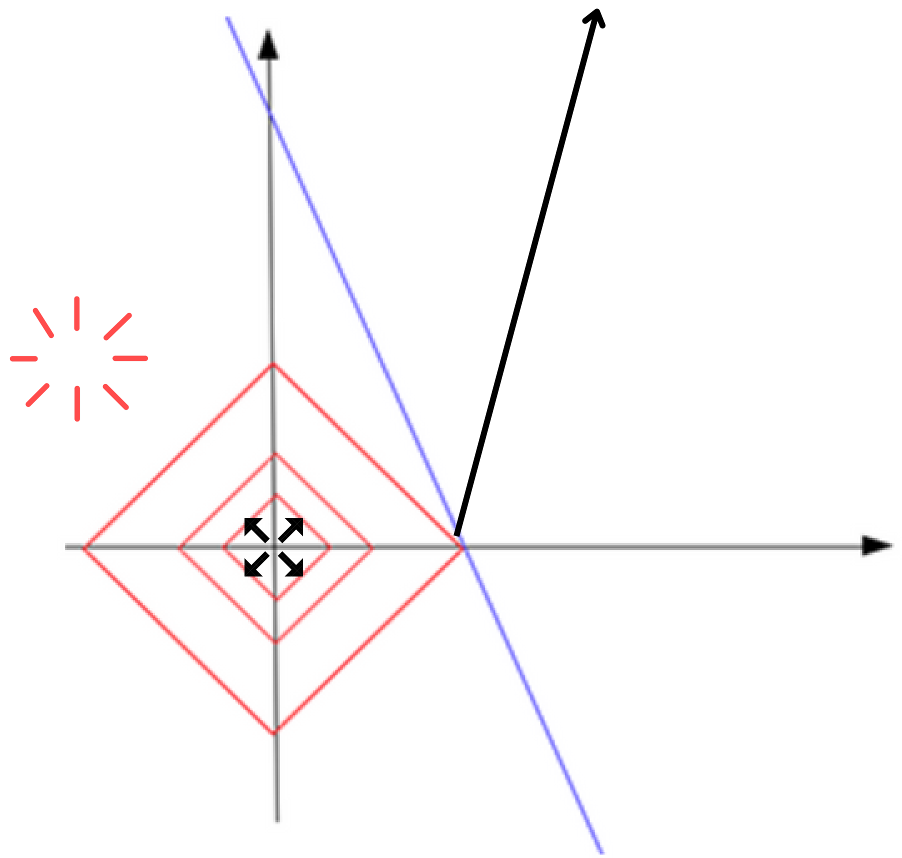
Sparse Point



Points which have L1 norm equal to some constant **c**

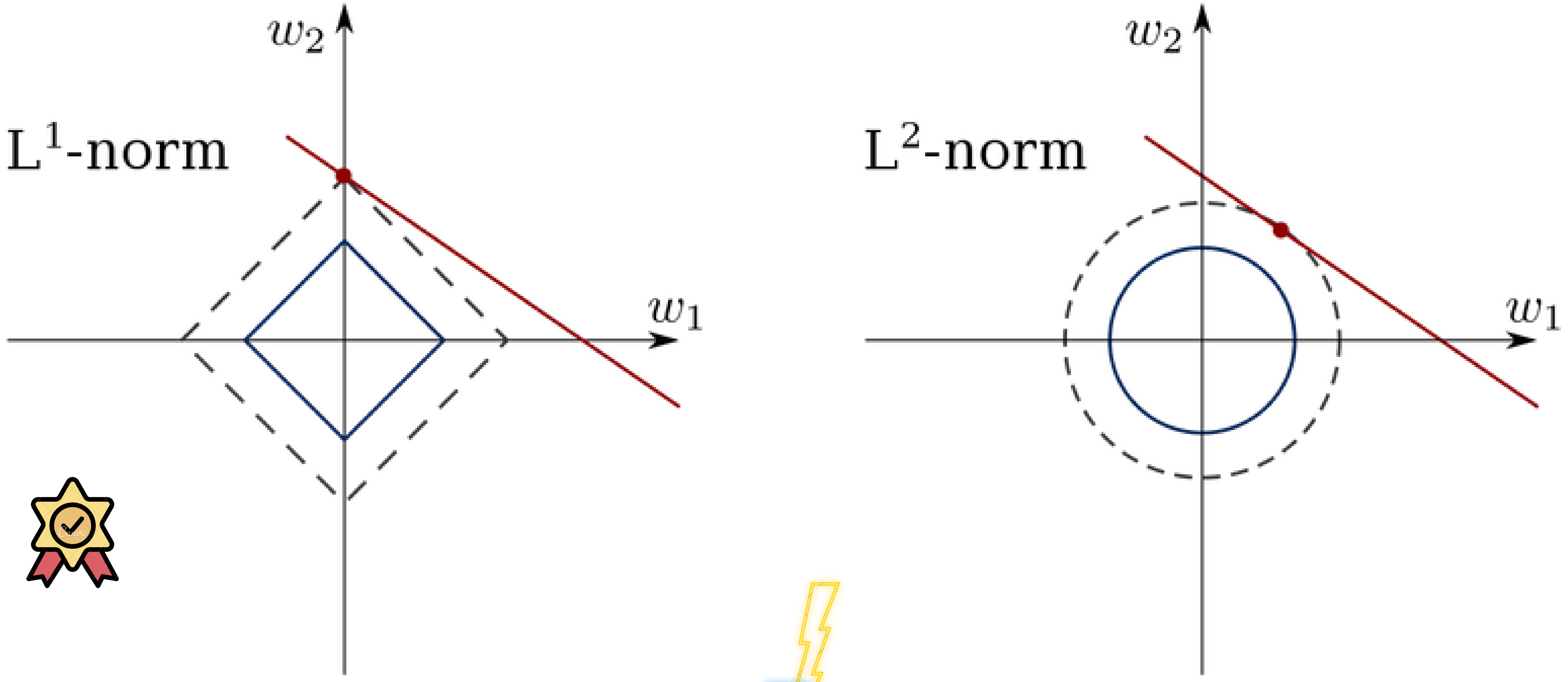
$$|a|+|b|=c$$

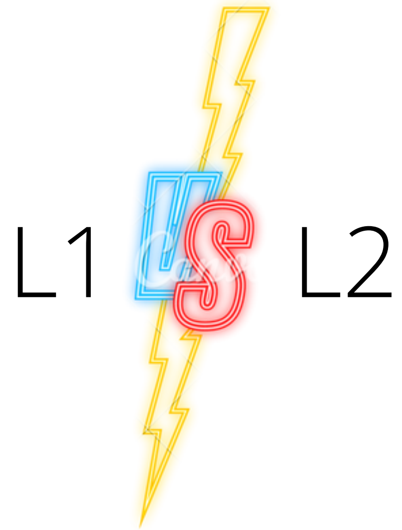
Our Sparse Solution



Value of **c** is increased till it touches the line

Inducing Sparsity using L1 Norm





Principal component pursuit

$$\min_{L,S} \; \|L\|_* + \lambda||S||_1 \text{ subject to } L + S = X.$$

Nuclear Norm

L1- Norm

here $\lambda = 1/\sqrt{\max(n, m)}$

n,m are dimensions of X

regularization parameter

Vector & Matrix Derivatives

Inner Product

Vectors

$$\mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}, \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

$$\frac{\partial}{\partial \mathbf{b}} \mathbf{a}^t \mathbf{b} = \frac{\partial}{\partial \mathbf{b}} (a_1 b_1 + a_2 b_2 + a_3 b_3)$$

$$\frac{\partial}{\partial \mathbf{b}} \mathbf{a}^t \mathbf{b} = \begin{bmatrix} \frac{\partial}{\partial b_1}(a_1 b_1) \\ \frac{\partial}{\partial b_2}(a_2 b_2) \\ \frac{\partial}{\partial b_3}(a_3 b_3) \end{bmatrix}$$

$$\frac{\partial}{\partial \mathbf{b}} \mathbf{a}^t \mathbf{b} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \mathbf{a}$$

Inner Product

Matrices

A = $\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$ $A^T = \begin{bmatrix} a_{11} & a_{21} \\ a_{12} & a_{22} \end{bmatrix}$ B = $\begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}$

$$A^T B = \begin{bmatrix} a_{11}b_{11} + a_{21}b_{21} & a_{11}b_{12} + a_{21}b_{22} \\ a_{12}b_{11} + a_{22}b_{21} & a_{12}b_{12} + a_{22}b_{22} \end{bmatrix}$$

$$Tr(A^T B) = a_{11}b_{11} + a_{21}b_{21} + a_{12}b_{12} + a_{22}b_{22}$$

Inner Product

Matrices

Derivative

$$\frac{\partial \langle A, B \rangle}{\partial B} = \begin{bmatrix} \frac{\partial}{\partial b_{11}} a_{11} b_{11} & \frac{\partial}{\partial b_{12}} a_{12} b_{12} \\ \frac{\partial}{\partial b_{21}} a_{21} b_{21} & \frac{\partial}{\partial b_{22}} a_{22} b_{22} \end{bmatrix}$$

$$= \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = \mathbf{A}$$

L1 Norm

Vectors

$$\mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \qquad \left\|\mathbf{a}\right\|_1 = \sum_i |a_i| = |a_1| + |a_2| + |a_3|$$

$$\frac{\partial \left\|\mathbf{a}\right\|_1}{\partial \mathbf{a}} = \begin{bmatrix} \frac{\partial |a_1|}{\partial a_1} \\ \frac{\partial |a_2|}{\partial a_2} \\ \frac{\partial |a_3|}{\partial a_3} \end{bmatrix} = \begin{bmatrix} sign(a_1) \\ sign(a_2) \\ sign(a_3) \end{bmatrix} = sign(\mathbf{a})$$

L1 Norm

Matrices

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

$$\|\mathbf{A}\|_1 = \sum_{ij} |a_{ij}| = |a_{11}| + |a_{12}| + |a_{21}| + |a_{22}|$$

"Entry-wise" L1 Norm

$$\frac{\partial \|\mathbf{A}\|_1}{\partial \mathbf{A}} = \begin{bmatrix} \frac{\partial |a_{11}|}{\partial a_{11}} & \frac{\partial |a_{12}|}{\partial a_{12}} \\ \frac{\partial |a_{21}|}{\partial a_{21}} & \frac{\partial |a_{22}|}{\partial a_{22}} \end{bmatrix} = \begin{bmatrix} sign(a_{11}) & sign(a_{12}) \\ sign(a_{21}) & sign(a_{22}) \end{bmatrix} = sign(\mathbf{A})$$

Frobenius Norm

$$A = \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix}, B = \begin{bmatrix} b_1 & b_2 \\ b_3 & b_4 \end{bmatrix}$$

$$\|A+B\|_F^2 = \sum_{i,j} (A+B)_{i,j}^2 = (a_1+b_1)^2 + (a_2+b_2)^2 + (a_3+b_3)^2 + (a_4+b_4)^2$$

$$\frac{\partial \|A+B\|_F^2}{\partial B} = \begin{bmatrix} \frac{\partial (a_1+b_1)^2}{\partial b_1} & \frac{\partial (a_2+b_2)^2}{\partial b_2} \\ \frac{\partial (a_3+b_3)^2}{\partial b_3} & \frac{\partial (a_4+b_4)^2}{\partial b_4} \end{bmatrix} = \begin{bmatrix} 2(a_1+b_1) & 2(a_2+b_2) \\ 2(a_3+b_3) & 2(a_4+b_4) \end{bmatrix}$$

$$2(A+B)$$

Principal component pursuit by Alternating directions

Augmented Lagrangian

$$\min_{L,S} ||L||_* + \lambda ||S||_1$$

subject to

$$X = L + S$$

$$\mathcal{L}(L, S, Y) = ||L||_* + \lambda ||S||_1 - \langle Y, X - L - S \rangle + \frac{\mu}{2} ||X - L - S||_F^2$$

$$X \in \mathbb{R}^{p \times n}, L \in \mathbb{R}^{p \times n}, S \in \mathbb{R}^{p \times n}$$

Y is a matrix of Lagrangian multipliers, $Y \in \mathbb{R}^{p \times n}$

$\frac{\mu}{2}$ is Augmented lagrangian term

Update L

$$\mathcal{L}(L) = ||L||_* - \langle Y, -L \rangle + \frac{\mu}{2}||X - L - S||_F^2$$

$$L^{k+1} = \underset{L}{argmin}\, ||L||_* + \left\langle Y^k, L \right\rangle + \frac{\mu}{2}\left|\left|X - L - S^k\right|\right|_F^2$$

$$\underset{L}{argmin} f(L) = ||L||_* + \left\langle Y^k, L \right\rangle + \frac{\mu}{2}\left|\left|X - L - S^k\right|\right|_F^2$$

$$\nabla f(L^*) = 0 \qquad 0 = \partial(||L||_*) + Y^k + \mu(L - X + S^k)$$

SVT Operator theorem

The solution to the following optimization problem

$$\mathbf{X}^* = \underset{x}{argmin} \frac{1}{2} ||X - Y||_F^2 + \lambda ||X||_*$$

$$\mathbf{X}^* = SVT_\lambda(\mathbf{Y}) := \mathbf{U}[\Sigma - \lambda \mathbf{I}]_+ \mathbf{V}^T$$

Where $\mathbf{U}\Sigma\mathbf{V}^T = SVD(\mathbf{Y})$ and $[.]_+ = max(., 0)$

Proved using using von Neumann trace inequality
Alternative proof of the SVT operator theorem - using von Neumann trace inequality (angms.science)

Equivalent Function

$$L^{k+1} = \underset{L}{argmin} \, ||L||_* + \frac{\mu}{2} \left|\left| X - L - S^k + \frac{Y^k}{\mu} \right|\right|_F^2$$

$$\nabla f(L^*) = 0$$

$$0 = \partial(||L||_*) + Y^k + \mu(L - X + S^k)$$

$$\boxed{\begin{array}{l} \underset{L}{argmin} f(L) = ||L||_* + \langle Y^k, L \rangle + \frac{\mu}{2} ||X - L - S^k||_F^2 \\ 0 = \partial(||L||_*) + Y^k + \mu(L - X + S^k) \end{array}}$$

Update L

$$L^{k+1} = \underset{L}{argmin} \, ||L||_* + \frac{\mu}{2} \left|\left| X - L - S^k + \frac{Y^k}{\mu} \right|\right|_F^2$$

$$\boxed{L^{k+1} = SVT_{\frac{1}{\mu}}(X - S^k + \frac{Y^k}{\mu})}$$

Update S

$$\mathcal{L}(S) = \lambda\,||S||_1 - \langle Y, -S\rangle + \frac{\mu}{2}\,||X - L - S||_F^2$$

$$S^{k+1} = \underset{S}{argmin}\ \lambda\,||S||_1 + \left\langle Y^k, S\right\rangle + \frac{\mu}{2}\,||X - L^{k+1} - S||_F^2$$

$$\underset{S}{argmin}\ f(S) = \ \lambda\,||S||_1 + \left\langle Y^k, S\right\rangle + \frac{\mu}{2}\,||X - L^{k+1} - S||_F^2$$

$$0 = \lambda\partial(||S||_1) + Y^k + \mu(-X + L^{k+1} + S)$$

Update S

$$0 = \lambda \partial(||S||_1) + Y^k + \mu(X - L^{k+1} - S)$$

$$sign(S) + \frac{1}{\frac{\lambda}{\mu}}(S - (X - L^{k+1} + \frac{Y^k}{\mu})) = 0$$

Shrinkage function

$$\underset{s}{min} f(s) = ||s||_1 + \frac{1}{2\tau} ||s - x||_2^2$$

$$0 = sign(s) + \frac{1}{\tau}(s - x)$$

$$sign(S) + \frac{1}{\frac{\lambda}{\mu}}(S - (X - L^{k+1} + \frac{Y^k}{\mu})) = 0$$

$$\boxed{\begin{gathered} s^* = x - \tau sign(x) \\ S_\tau(x) = \begin{cases} x - \tau & \text{if } x > 0, x > \tau \\ x + \tau & \text{if } x < 0, |x| > \tau \\ 0 & \text{if } \ |x| \leq \tau \end{cases} \end{gathered}}$$

$$S_\tau(x) = sign(x) max(|x - \tau|, 0)$$

Update S

$$sign(S) + \frac{1}{\frac{\lambda}{\mu}}(S - (X - L^{k+1} + \frac{Y^k}{\mu})) = 0$$

$$\boxed{S^{k+1} = \mathbf{S}_{\frac{\lambda}{\mu}}(X - L^{k+1} + \frac{Y^k}{\mu})}$$

Update Y

$$\frac{\partial \mathcal{L}}{\partial Y} = X - L - S$$

$$Y^{k+1} = Y^k + \textit{step-size} \times \textit{gradient of } \mathcal{L} \textit{ w.r.t } Y$$

$$\boxed{Y^{k+1} = Y^k + \mu(X - L^{k+1} - S^{k+1})}$$

$$\mu = \frac{mn}{4\left\|X\right\|_1}$$

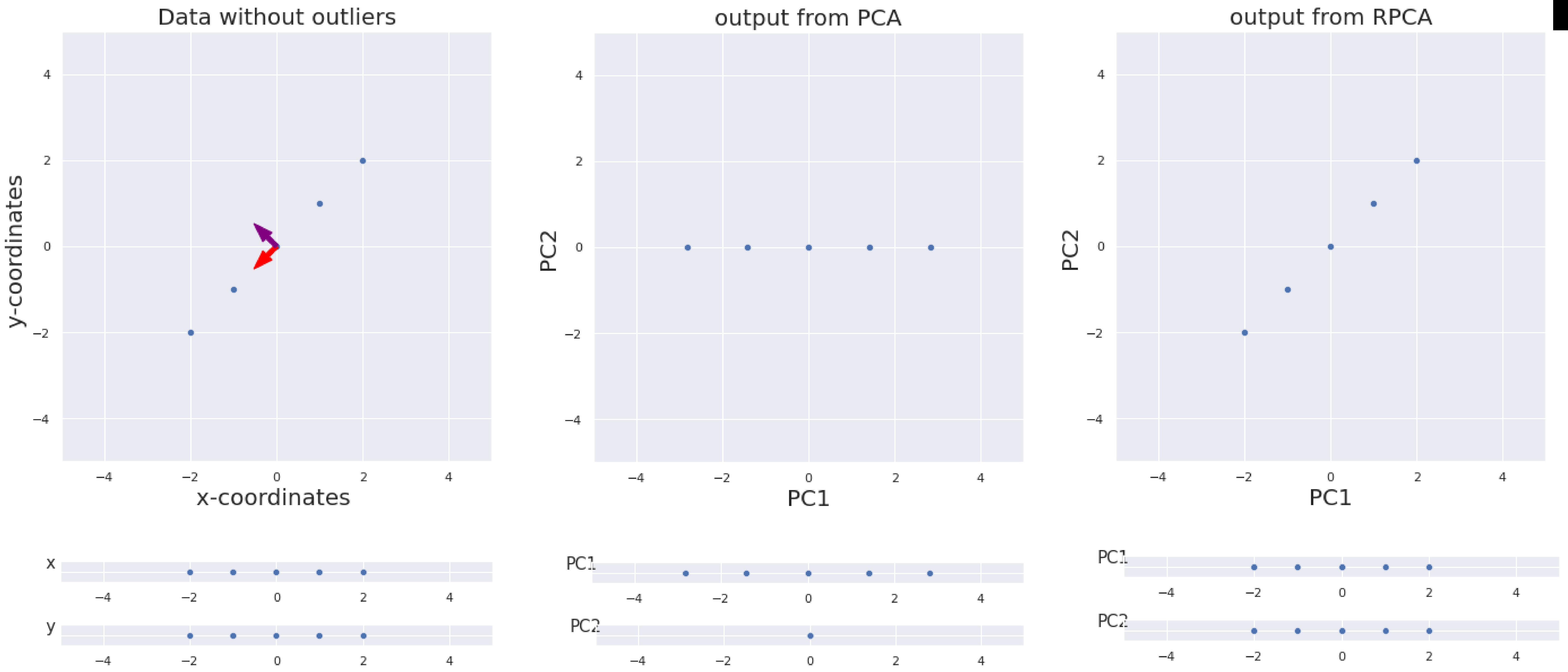
Where m,n are dimensions of X

X. Yuan and J. Yang. Sparse and low-rank mat decomposition via alternating direction methods.preprint, 2009.

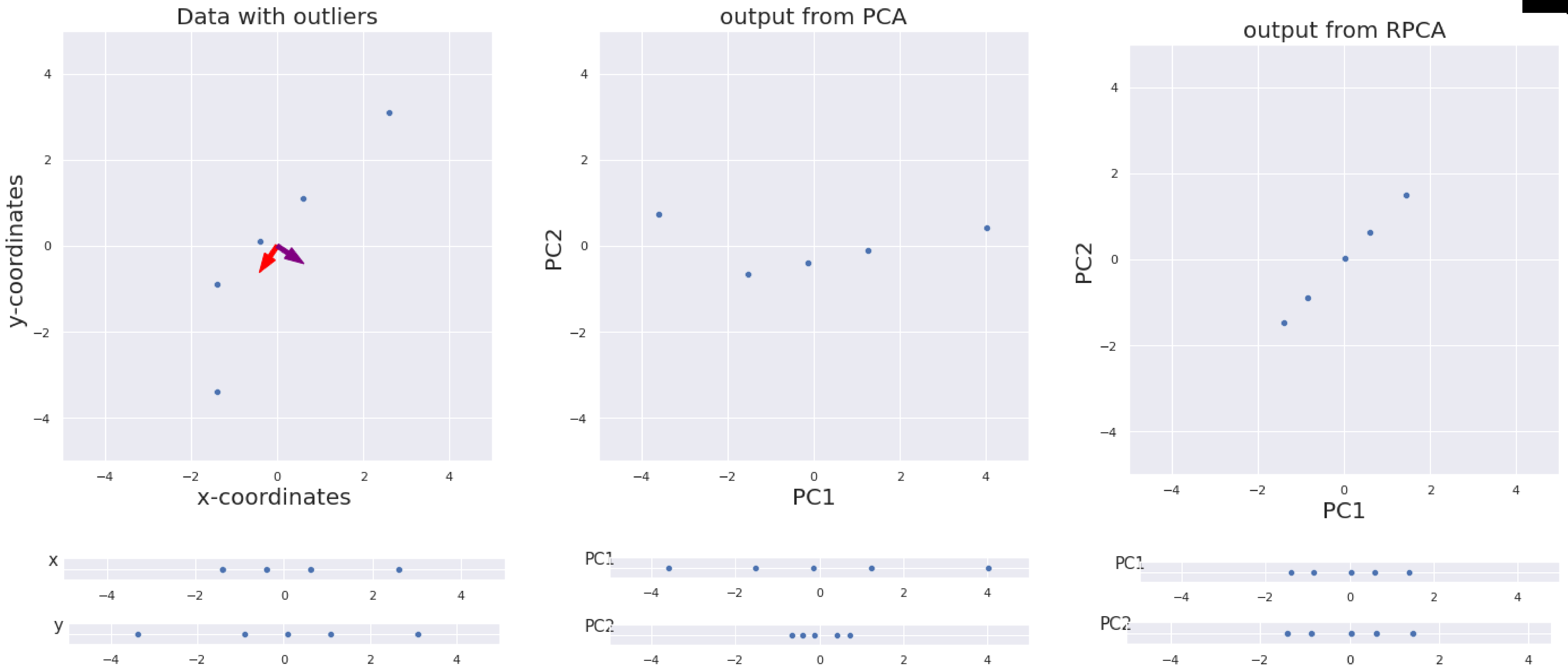
$$\lambda = \frac{1}{\sqrt{max(m,n)}}$$

Where m,n are dimensions of X

RPCA Without Outliers



RPCA With Outliers



Complete Algorithm

Algorithm 1 Principal Component Pursuit by Alternating Directions

Require: $S_0 = Y_0 = 0, \mu > 0$

while not converged **do**

compute $L^{k+1} = SVT_{\frac{1}{\mu}}(X - S^k + \frac{Y^k}{\mu})$

compute $S^{k+1} = S_{\frac{\lambda}{\mu}}(X - L^{k+1} + \frac{Y^k}{\mu})$

compute $Y^{k+1} = Y^k + \mu(X - L^{k+1} - S^{k+1})$

end while

return L, S

Application

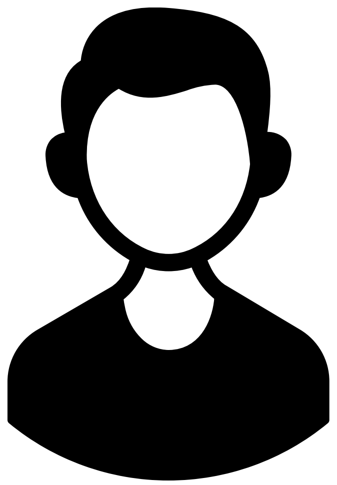
Yale dataset of faces

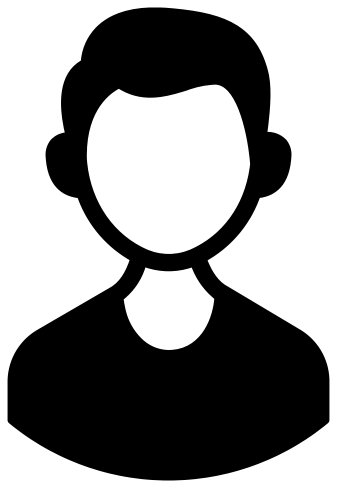
Our Data

2 individuals

64 + 62 pictures

each picture 192 x 168 = 32,256 pixels



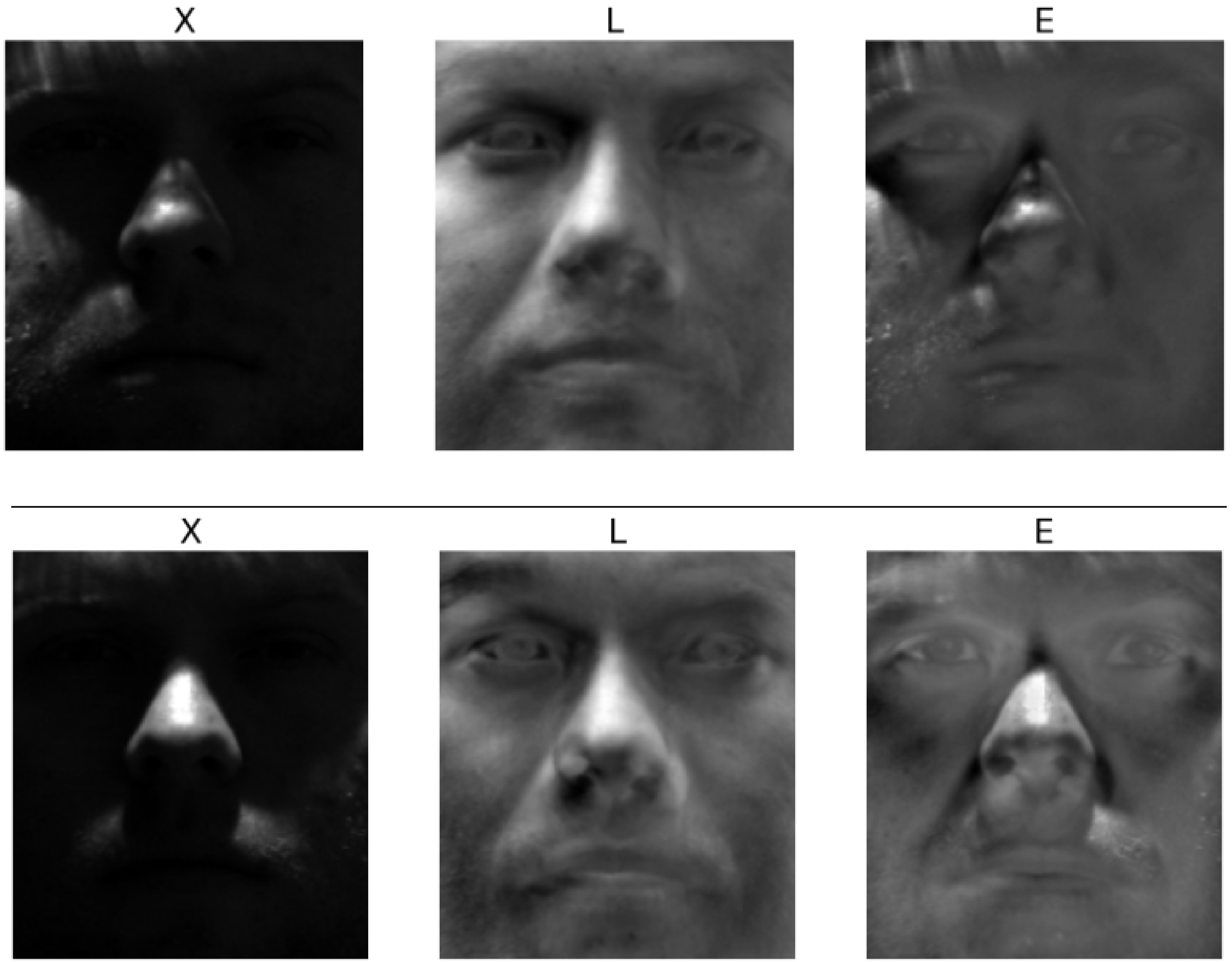


126 , 192 , 168

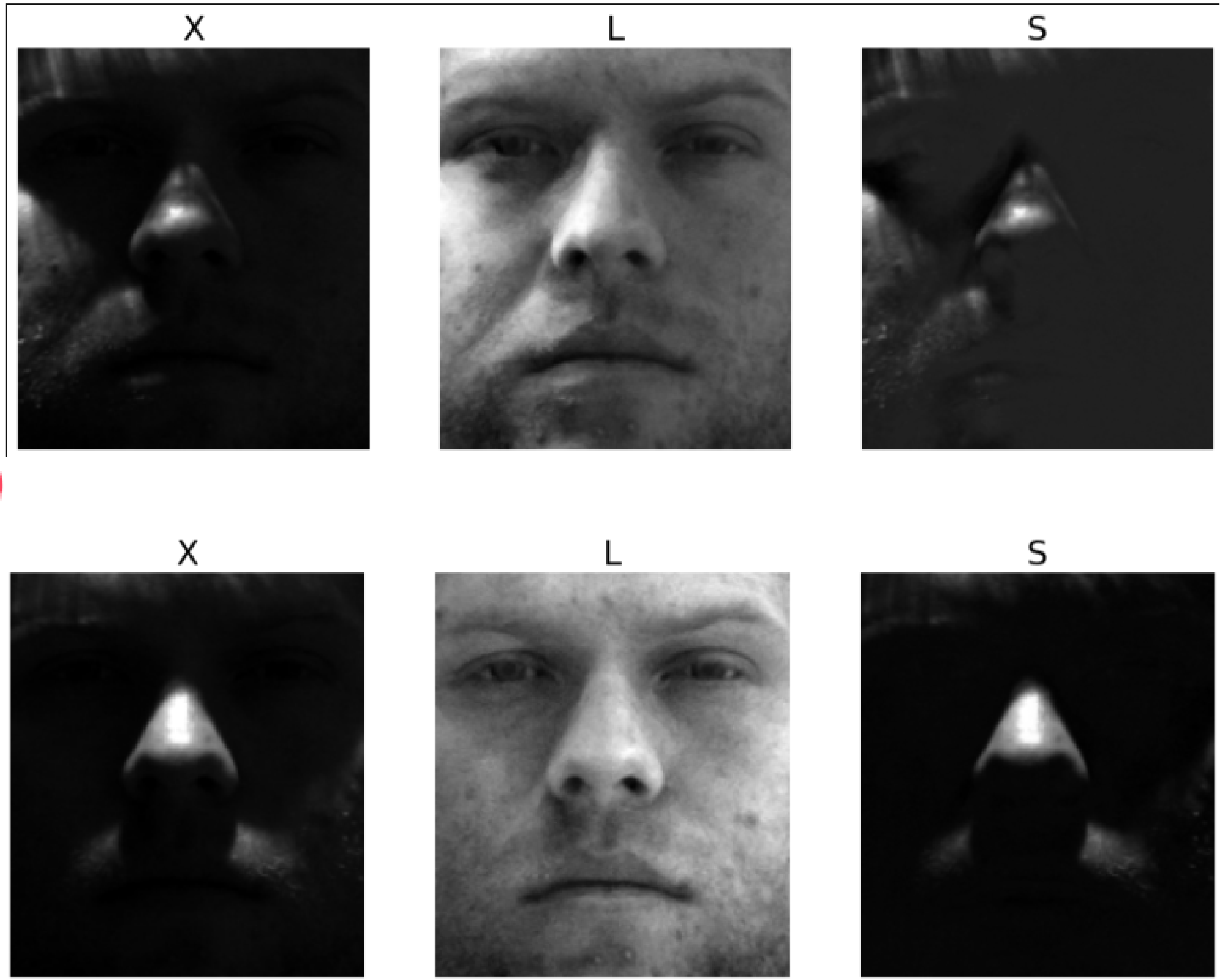
32256, 126

126 columns where each column is an image

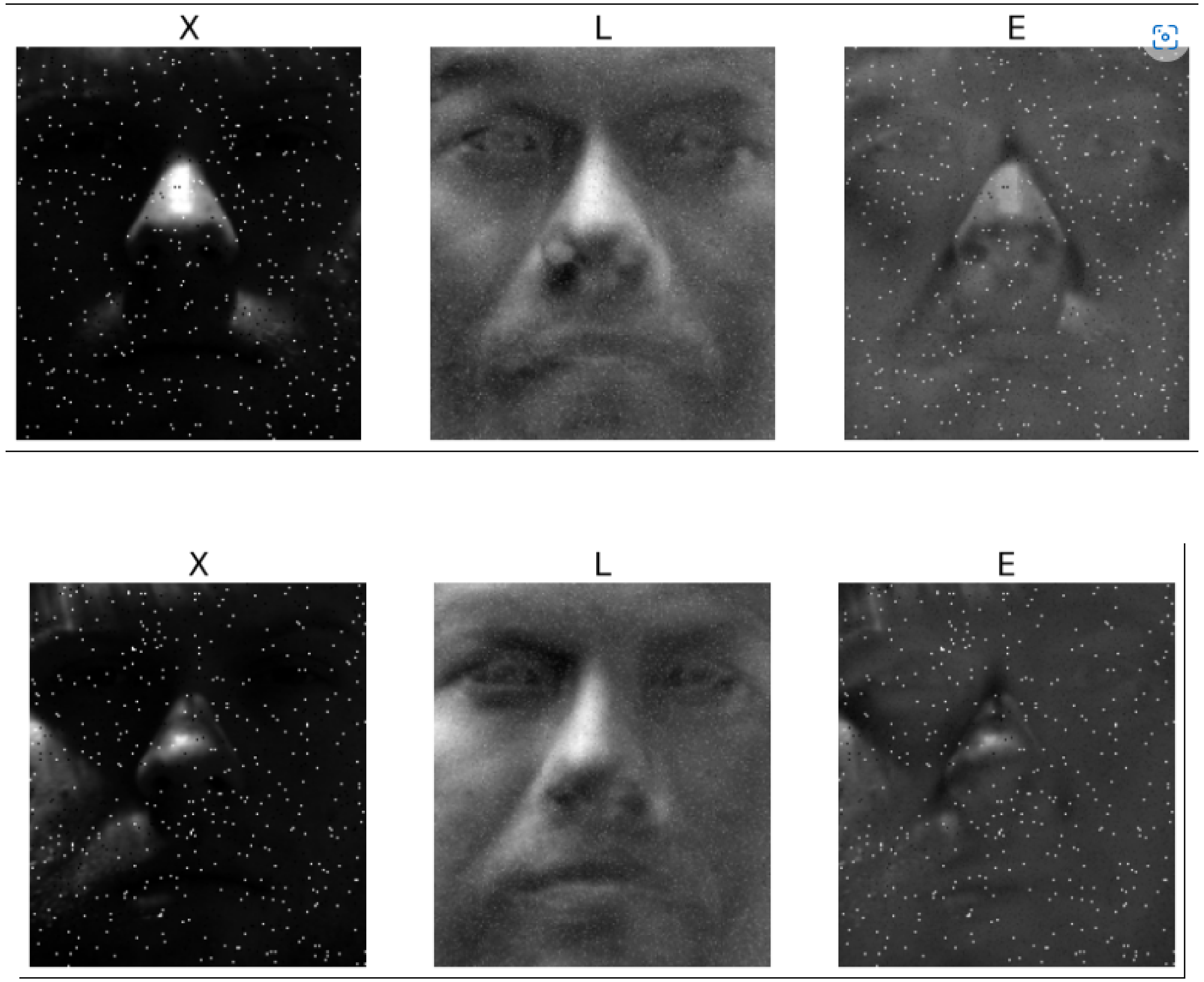
PCA



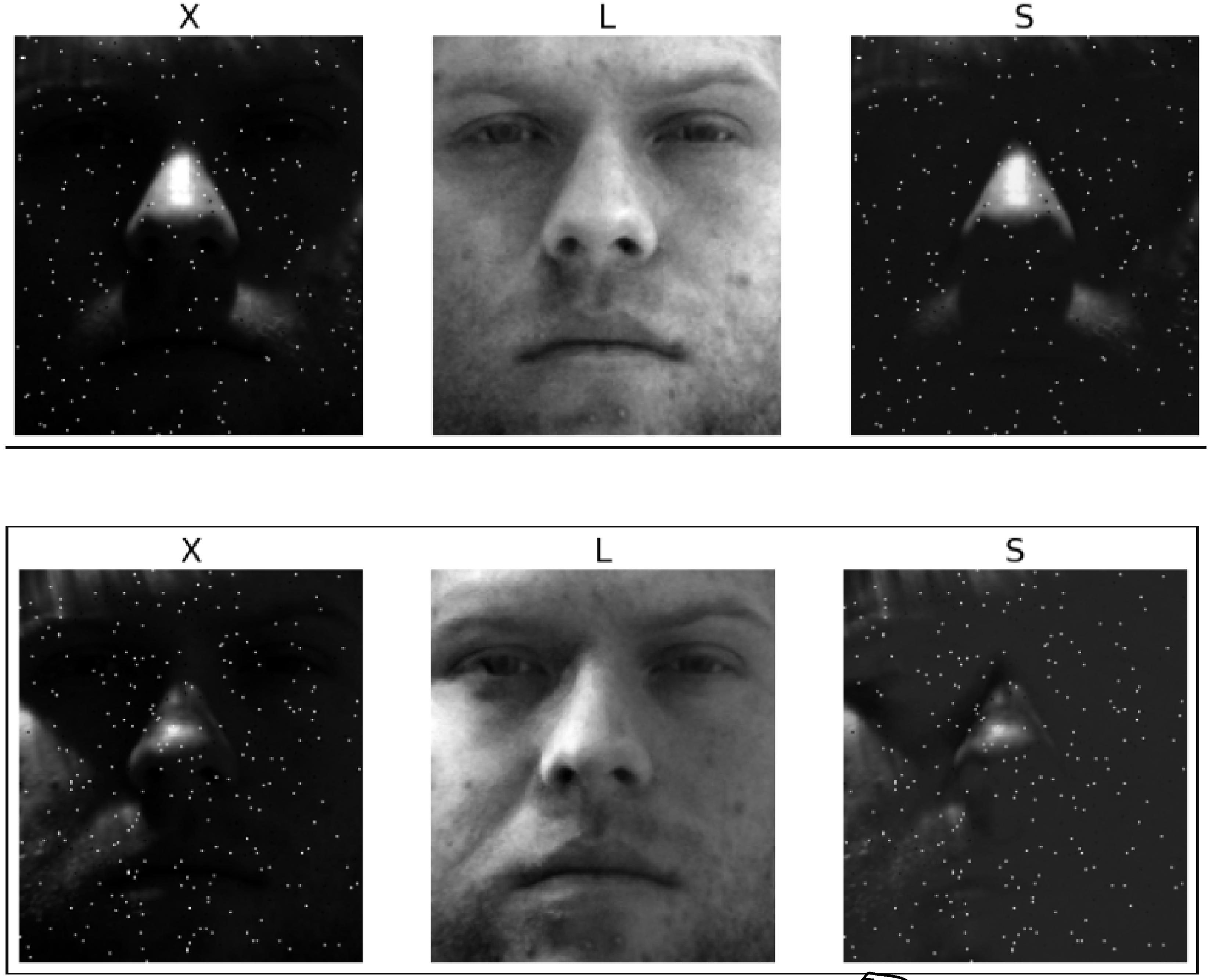
RPCA



PCA

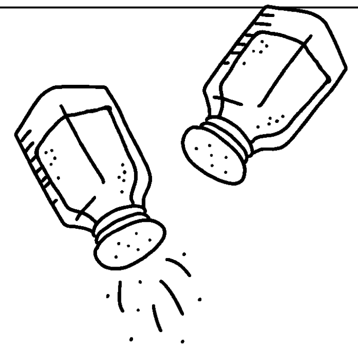


RPCA

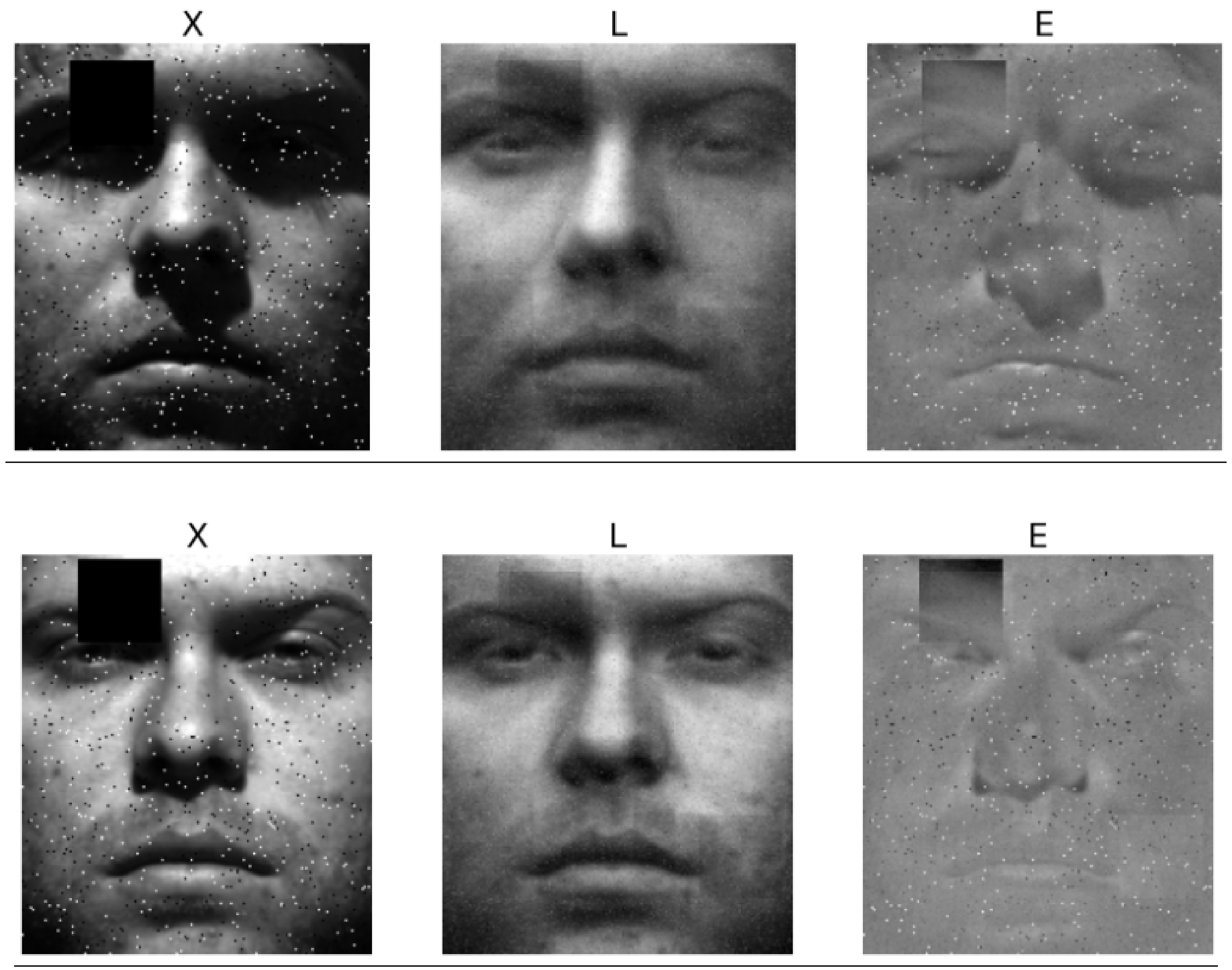


VS

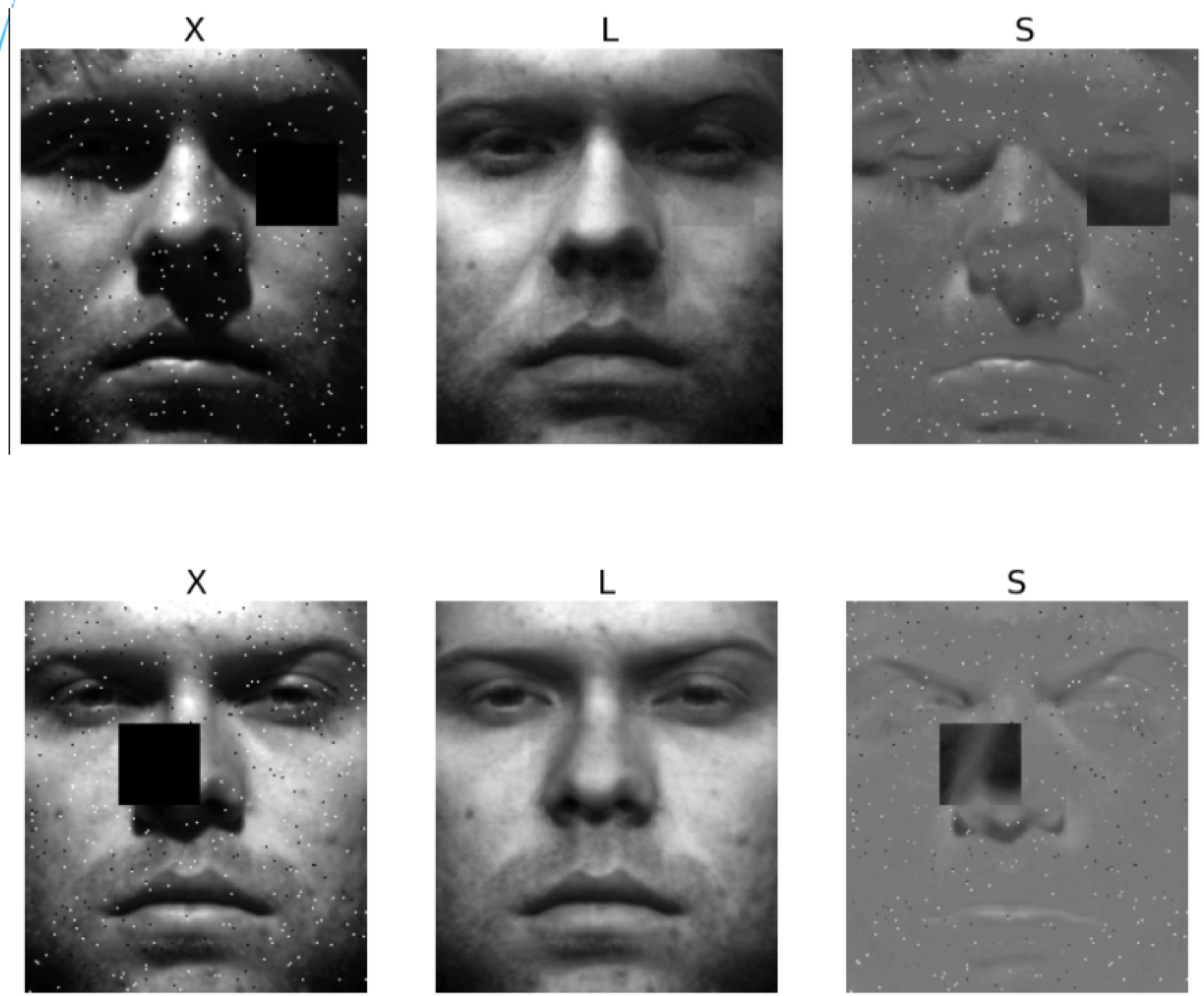
Images with Salt and Pepper Noise



PCA



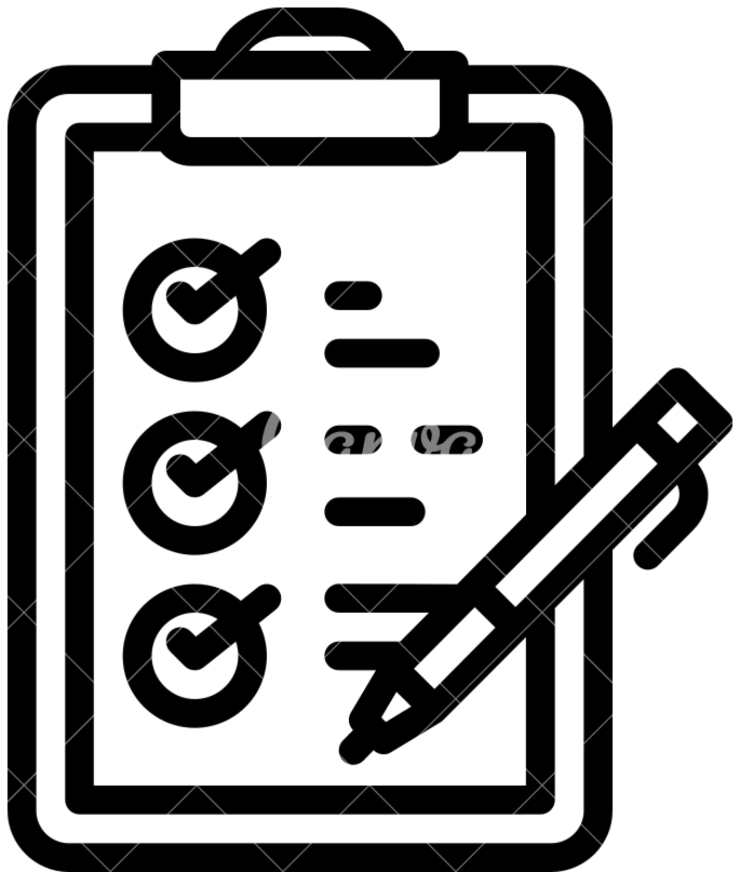
RPCA



Images with added Patches

Conclusion

- Both PCA and RPCA are good techniques for matrix decomposition.
- RPCA helps to tackle the 'Outlier Problem'.



Thank you