ROBUST 2018

25.1.2018

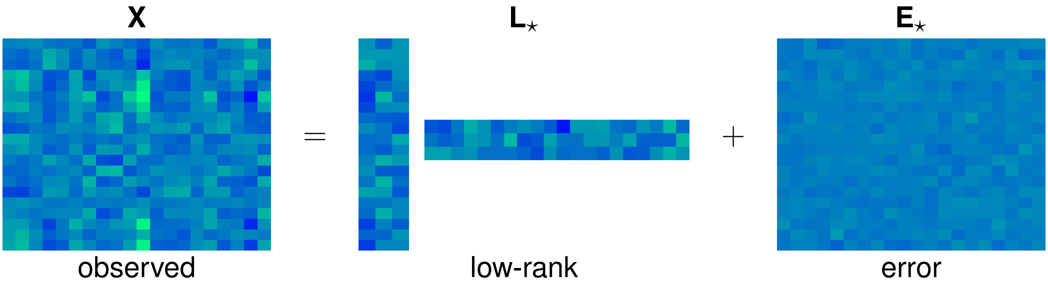
Robust Principal Component Analysis

Tomáš Masák
Joint work with Christian Kümmerle and Felix Krahmer

Technische Universität München

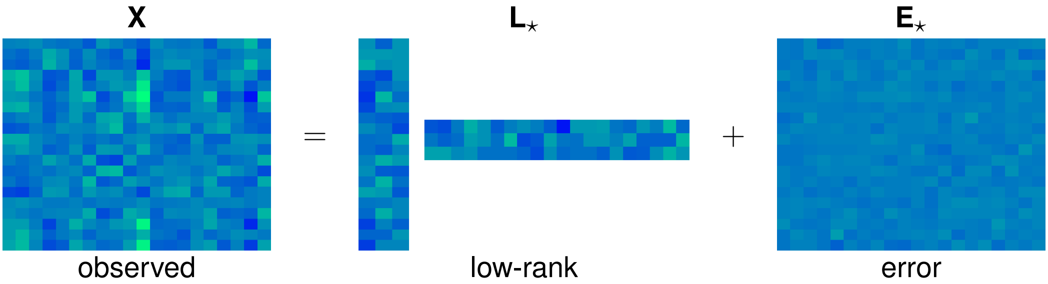
Principal Component Analysis (PCA)

Standard PCA

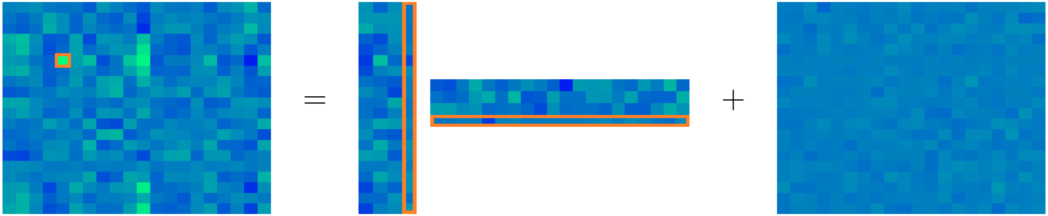


Principal Component Analysis (PCA)

Standard PCA

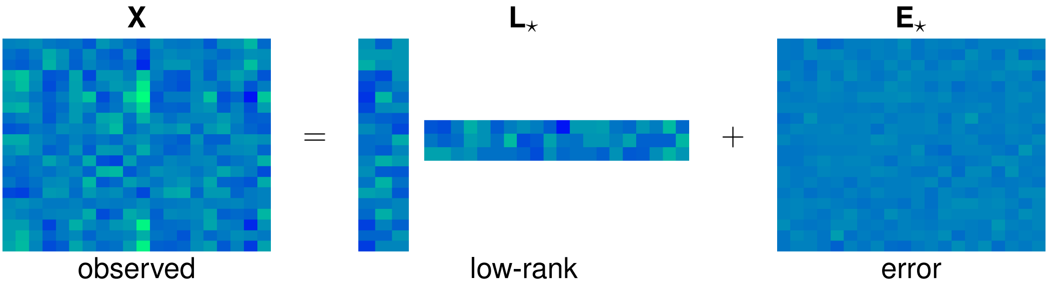


Standard PCA with outliers

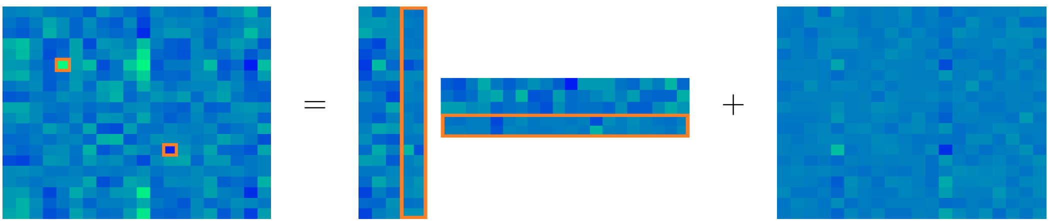


Principal Component Analysis (PCA)

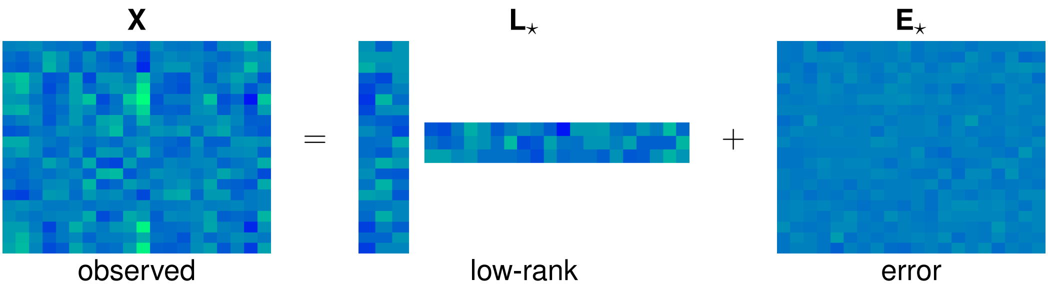
Standard PCA



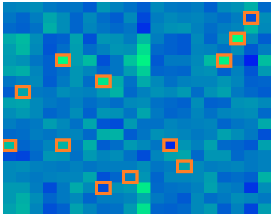
Standard PCA with outliers



Standard PCA



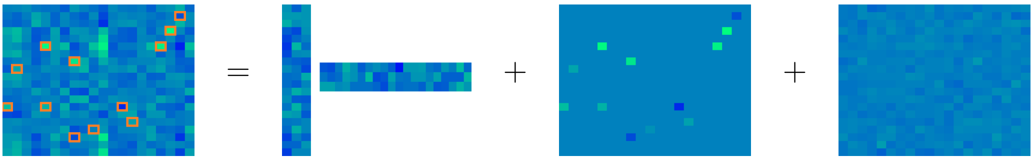
Standard PCA with outliers



$= \quad ? \quad + \quad ?$

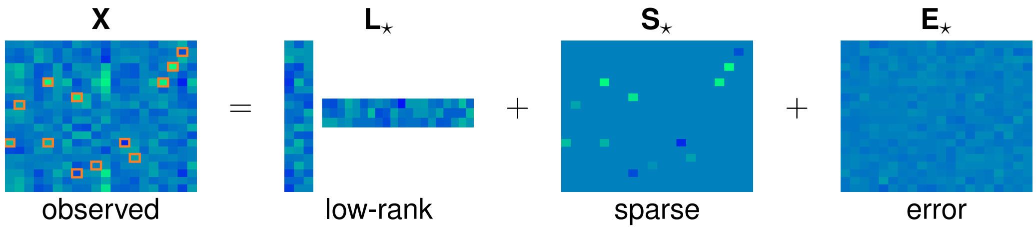
Robust PCA:

$\mathbf{X} = \mathbf{L}_\star + \mathbf{S}_\star + \mathbf{E}_\star$

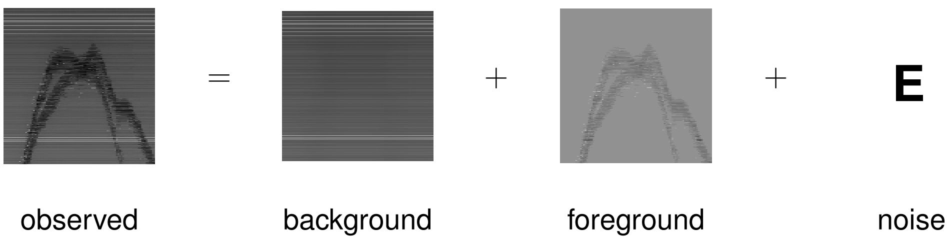


observed = low-rank + sparse + error

Robust PCA:

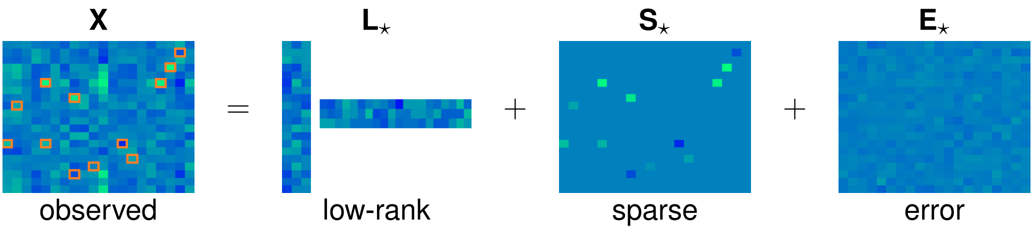


Application: video surveillance

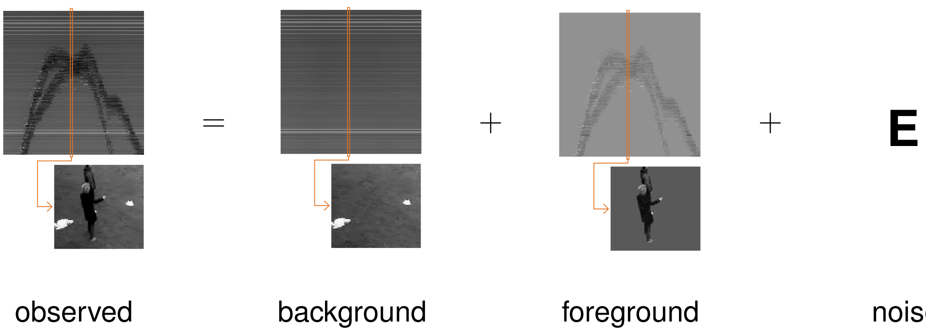


Robust PCA

Robust PCA:

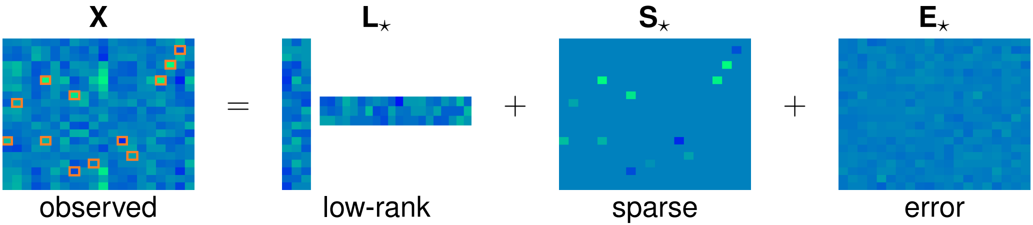


Application: video surveillance



Robust PCA

Robust PCA:



Application: video surveillance

$$= \qquad + \qquad + \quad \mathbf{E}$$

observed background foreground noise

Robust PCA

Problem: Given matrix $\mathbf{X} \in \mathbb{R}^{n_1 \times n_2}$, find a low-rank matrix $\mathbf{L}_\star$ and a sparse matrix $\mathbf{S}_\star$ such that

$$\begin{array}{rccccc} & \mathbf{X} & = & \mathbf{L}_\star & + & \mathbf{S}_\star \\ \text{df:} & n_1 n_2 & \geq & (n_1 + n_2) r & + & s \end{array}$$

where $r = \operatorname{rank} \mathbf{L}_\star$ and $s = \|(\mathbf{S}_\star)_{\text{vec}}\|_0$

Is the problem solveable?

Robust PCA

Problem: Given matrix $\mathbf{X} \in \mathbb{R}^{n_1 \times n_2}$, find a low-rank matrix $\mathbf{L}_\star$ and a sparse matrix $\mathbf{S}_\star$ such that

$$\begin{array}{rcccccc} & \mathbf{X} & = & \mathbf{L}_\star & + & \mathbf{S}_\star \\ \text{df:} & n_1 n_2 & \geq & (n_1 + n_2) r & + & s \end{array}$$

where $r = \operatorname{rank} \mathbf{L}_\star$ and $s = \|(\mathbf{S}_\star)_{\text{vec}}\|_0$

Is the problem solveable?

Natural optimization formulation:

$$\begin{aligned} (\widehat{\mathbf{L}}, \widehat{\mathbf{S}}) \quad := \quad & \underset{\mathbf{L},\mathbf{S}}{\arg\min} \quad \operatorname{rank} \mathbf{L} + \lambda \|\mathbf{S}_{\text{vec}}\|_0 \\ & \text{s.t.} \qquad \mathbf{X} = \mathbf{L} + \mathbf{S} \end{aligned}$$

It is **NP-hard**.

Robust PCA

Problem: Given matrix $\mathbf{X} \in \mathbb{R}^{n_1 \times n_2}$, find a low-rank matrix $\mathbf{L}_\star$ and a sparse matrix $\mathbf{S}_\star$ such that

$$\begin{array}{ccccccc} & \mathbf{X} & = & \mathbf{L}_\star & + & \mathbf{S}_\star \\ \text{df:} & n_1 n_2 & \geq & (n_1 + n_2) r & + & s \end{array}$$

where $r = \operatorname{rank} \mathbf{L}_\star$ and $s = \|(\mathbf{S}_\star)_{\text{vec}}\|_0$

Is the problem solveable?

Natural optimization formulation:

$$\begin{aligned} (\widehat{\mathbf{L}}, \widehat{\mathbf{S}}) := & \operatorname*{arg\,min}_{\mathbf{L},\mathbf{S}} \quad \operatorname{rank} \mathbf{L} + \lambda \|\mathbf{S}_{\text{vec}}\|_0 \\ & \text{s.t.} \quad \mathbf{X} = \mathbf{L} + \mathbf{S} \end{aligned}$$

It is **NP-hard**.

The seminal paper of Candés et al. (2011) showed that (given certain assumptions) the separation is possible via a convex program

$$\begin{aligned} (\widehat{\mathbf{L}}, \widehat{\mathbf{S}}) := & \operatorname*{arg\,min}_{\mathbf{L},\mathbf{S}} \quad \operatorname{rank} \|\mathbf{L}\|_* + \lambda \|\mathbf{S}_{\text{vec}}\|_1 \\ & \text{s.t.} \quad \mathbf{X} = \mathbf{L} + \mathbf{S} \end{aligned}$$

Issues with the convex approach:

- relatively slow
- mediocre performance (especially in practical applications, where the assumptions are typically broken)

Robust PCA

Issues with the convex approach:

- relatively slow
- mediocre performance (especially in practical applications, where the assumptions are typically broken)

$\Rightarrow$ new algorithms emerged in the past few years, most of them non-convex

We also propose a non-convex algorithm...

Proposed algorithm

Derivation of the algorithm in a nutshell:

1. Smooth the **non-convex objective** to become **differentiable**.
2. Express the derivative in a **special form**.
3. Solve the system of **non-linear equations** arising from the first order optimality conditions via a **fixed point scheme**.

The resulting algorithm is an instance of Iteratively Reweighted Least Squares (**IRLS**) method.

Proposed algorithm

Derivation of the algorithm in a nutshell:

1. Smooth the **non-convex objective** to become **differentiable**.
2. Express the derivative in a **special form**.
3. Solve the system of **non-linear equations** arising from the first order optimality conditions via a **fixed point scheme**.

The resulting algorithm is an instance of Iteratively Reweighted Least Squares (**IRLS**) method.

Where is the novelty?

– special form of the derivative $\rightarrow$ a **local quadratic convergence rate**
– a competitive performance

Proposed algorithm

Our algorithm is **not**

1. fastest – the algorithm of Yi et al. (2016) attains the time complexity of the standard PCA and thus is hard to beat
2. statistically most accurate – the algorithm of Oh et al. (2016) is hard to beat
3. numerically most accurate

Proposed algorithm

Our algorithm is **not**

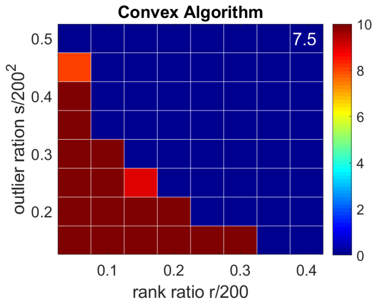
1. fastest – the algorithm of Yi et al. (2016) attains the time complexity of the standard PCA and thus is hard to beat
2. statistically most accurate – the algorithm of Oh et al. (2016) is hard to beat
3. numerically most accurate – actually, **it is**

Small Comparison

Take $\mathbf{X} = \mathbf{A}\mathbf{B}^\top$ with $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{200 \times \mathbf{r}}$ and replace **s** random entries by random corruptions. Algorithm succeeds if $\|\widehat{\mathbf{L}} - \mathbf{A}\mathbf{B}^\top\|_F / \|\mathbf{A}\mathbf{B}^\top\|_F < 0.01$.

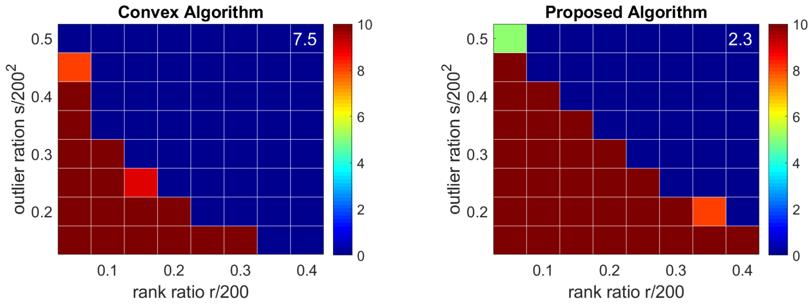
Small Comparison

Take $\mathbf{X} = \mathbf{AB}^\top$ with $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{200 \times \mathbf{r}}$ and replace **s** random entries by random corruptions. Algorithm succeeds if $\|\widehat{\mathbf{L}} - \mathbf{AB}^\top\|_F / \|\mathbf{AB}^\top\|_F < 0.01$.

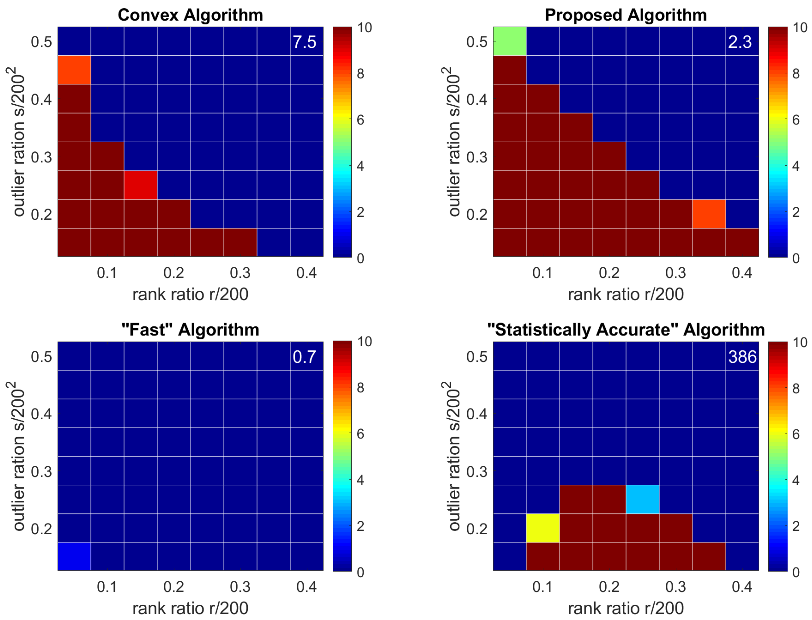


Small Comparison

Take $\mathbf{X} = \mathbf{A}\mathbf{B}^\top$ with $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{200 \times \mathbf{r}}$ and replace **s** random entries by random corruptions. Algorithm succeeds if $\|\widehat{\mathbf{L}} - \mathbf{A}\mathbf{B}^\top\|_F / \|\mathbf{A}\mathbf{B}^\top\|_F < 0.01$.



Take $\mathbf{X} = \mathbf{A}\mathbf{B}^\top$ with $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{200 \times \mathbf{r}}$ and replace **s** random entries by random corruptions. Algorithm succeeds if $\|\widehat{\mathbf{L}} - \mathbf{A}\mathbf{B}^\top\|_F / \|\mathbf{A}\mathbf{B}^\top\|_F < 0.01$.



Conclusions

- A new algorithm for Robust PCA proposed.
  - the only competitive algorithm among the IRLS class
  - the only algortihm with super-linear convergence rate
  - the only algorithm uniformly outperforming the convex approach (disclaimer: subjective)
- The local quadratic convergence rate proved.

References

- Candès, E. J., Li, X., Ma, Y., & Wright, J. (2011). Robust principal component analysis?. *Journal of the ACM (JACM)*, 58(3), 11.
- Oh, T. H., Matsushita, Y., Kweon, I., & Wipf, D. (2016). A pseudo-bayesian algorithm for robust PCA. In *Advances in Neural Information Processing Systems* (pp. 1390-1398).
- Yi, X., Park, D., Chen, Y., & Caramanis, C. (2016). Fast algorithms for robust PCA via gradient descent. In *Advances in neural information processing systems* (pp. 4152-4160).