Scalable Robust Principal Component Analysis using Grassmann Averages — Supplementary Material —

Søren Hauberg, Aasa Feragen, Raffi Enficiaud, and Michael J. Black

APPENDIX A

We do not have explicit energy expressions for all the different variants of RGA, but in this appendix we consider the energy optimized by TGA$(50\%, 1)$ with unit weights. As this algorithm relies on trimmed spherical averages, we first consider these.

A.1 Trimmed Averages on S^{D-1}

In Euclidean spaces, the per-pixel trimmed average can be written as the solution to the following minimization problem

$$\boldsymbol{\mu}_{\mathrm{Trim},\mathbb{R}^D}(\boldsymbol{x}_{1:N}) = \operatorname*{arg\,min}_{\boldsymbol{\mu}\in\mathbb{R}^D} \sum_{n=1}^{N}\sum_{d=1}^{D} t_{nd}(x_{nd}-\mu_d)^2, \tag{33}$$

where the trimming weights $t_{nd} \in \{0,1\}$ denote which elements are "trimmed away" and which are kept. For $P\%$ trimming we have

$$\bar{t} = \sum_{n=1}^{N} t_{nd} = N - \frac{2NP}{100} \qquad \forall d = 1,\ldots,D. \tag{34}$$

Note that $\bar{t}$ is the same for all dimensions. The well-known solution to this problem is

$$\boldsymbol{\mu}_{\mathrm{Trim},\mathbb{R}^D}(\boldsymbol{x}_{1:N}) = \frac{1}{\bar{t}}\bar{\boldsymbol{x}}, \tag{35}$$

where $\bar{\boldsymbol{x}} \in \mathbb{R}^D$ has elements

$$\bar{x}_d = \sum_{n=1}^{N} t_{nd}x_{nd}. \tag{36}$$

In the main paper, we consider extrinsic trimmed averages on the unit sphere. These can be similarly defined as

$$\boldsymbol{\mu}_{\mathrm{Trim},S^{D-1}}(\boldsymbol{u}_{1:N}) = \operatorname*{arg\,min}_{\boldsymbol{\mu}\in S^{D-1}} \sum_{n=1}^{N}\sum_{d=1}^{D} t_{nd}(u_{nd}-\mu_d)^2. \tag{37}$$

This constrained optimization problem has a simple closed-form solution:

Lemma 2. *Let $\boldsymbol{u}_n \in S^{D-1}$ and let the trimming weights t_{nd} be fixed, then Eq. 37 has solution*

$$\boldsymbol{\mu}_{\mathrm{Trim},S^{D-1}}(\boldsymbol{u}_{1:N}) = \frac{1}{\|\bar{\boldsymbol{u}}\|}\bar{\boldsymbol{u}}, \tag{38}$$

where $\bar{u}_d = \sum_{n=1}^{N} t_{nd}u_{nd}$ is the Euclidean trimmed average of the u_{nd}..

Proof: The results follows by straight-forward computations: We seek the minima of

$$f(\boldsymbol{\mu}) = \sum_{n=1}^{N}\sum_{d=1}^{D} t_{nd}(u_{nd}-\mu_d)^2 \tag{39}$$

subject to the constraint $\|\boldsymbol{\mu}\| = 1$. We write the constraints using a Lagrange-multiplier

$$\hat{f}(\boldsymbol{\mu},\lambda) = f(\boldsymbol{\mu}) + \lambda g(\boldsymbol{\mu}), \quad \text{where} \tag{40}$$

$$g(\boldsymbol{\mu}) = 1 - \boldsymbol{\mu}^{\intercal}\boldsymbol{\mu} = 1 - \sum_{d=1}^{D}\mu_d^2. \tag{41}$$

We evaluate derivatives as

$$\frac{\partial f}{\partial \mu_d} = -2\sum_{n=1}^{N} t_{nd}(u_{nd}-\mu_d) \tag{42}$$

$$= -2\sum_{n=1}^{N} t_{nd}u_{nd} + 2\sum_{n=1}^{N} t_{nd}\mu_d \tag{43}$$

$$= -2\left(\bar{u}_d - \bar{t}\mu_d\right). \tag{44}$$

$$\frac{\partial g}{\partial \mu_d} = -2\mu_d \tag{45}$$

$$\frac{\partial \hat{f}}{\partial \mu_d} = -2\left(\bar{u}_d - \bar{t}\mu_d\right) - 2\lambda\mu_d \tag{46}$$

$$= -2\left(\bar{u}_d - (\bar{t}-\lambda)\mu_d\right). \tag{47}$$

- *S. Hauberg is with the Section for Cognitive Systems at the Technical University of Denmark.*
  E-mail: `sohau@dtu.dk`
- *A. Feragen is with the Department of Computer Science at the University of Copenhagen, Denmark.*
  E-mail: `aasa@diku.dk`
- *R. Enficiaud is with the Software Workshop at the Max Planck Institute for Intelligent Systems in Tübingen, Germany.*
  E-mail: `raffi.enficiaud@tue.mpg.de`
- *M.J. Black is with the Perceiving Systems Department at the Max Planck Institute for Intelligent Systems in Tübingen, Germany.*
  E-mail: `black@tue.mpg.de`

Setting $\partial \hat{f}/\partial \mu_d = 0$ gives

$$\mu_d = \frac{\bar{u}_d}{\bar{t} - \lambda}. \tag{48}$$

We evaluate λ by setting $\partial \hat{f}/\partial \lambda = 0$:

$$\frac{\partial \hat{f}}{\partial \lambda} = 1 - \sum_{d=1}^{D} \mu_d^2 = 1 - \sum_{d=1}^{D} \frac{\bar{u}_d^2}{(\bar{t} - \lambda)^2} \tag{49}$$

$$= 1 - (\bar{t} - \lambda)^{-2} \|\bar{\boldsymbol{u}}\|^2 = 0 \Rightarrow \tag{50}$$

$$\bar{t} - \lambda = \pm \|\bar{\boldsymbol{u}}\|. \tag{51}$$

Combining Eq. 48 and 51 gives

$$\mu_d = \pm \frac{1}{\|\bar{\boldsymbol{u}}\|} \bar{u}_d. \tag{52}$$

The unknown sign is determined by evaluating f at both choices and picking the smaller option. □

The per-pixel trimmed spherical average, thus, has the closed-form solution given by the per-pixel trimmed Euclidean average projected onto the sphere. In the case of 50% trimming, the per-pixel trimmed average coincides with the per-pixel median, and it follows that

$$\boldsymbol{\mu}_{\text{Median}, S^{D-1}}(\boldsymbol{u}_{1:N}) = \underset{\boldsymbol{\mu} \in S^{D-1}}{\arg\min} \sum_{n=1}^{N} \sum_{d=1}^{D} |u_{nd} - \mu_d| \tag{53}$$

can be solved by the per-pixel median projected onto the unit-sphere.

A.2 The Energy Optimized by $\mathsf{TGA}(50\%, 1)$

Intuitively, $\mathsf{TGA}(50\%, 1)$ should find the "median subspace" spanned by the data. Indeed it optimizes the energy

$$\boldsymbol{\mu}_{\text{Median}}(\boldsymbol{u}_{1:N}) = \underset{\boldsymbol{\mu} \in [\boldsymbol{\mu}]}{\arg\min} \left\{ \sum_{n=1}^{N} \sum_{d=1}^{D} |u_{nd} - \mu_d| \right\}, \tag{54}$$

which can be intepreted as a pixel-wise median subspace.

At every step, the TGA algorithm updates the representatives $\boldsymbol{u}_n$ of the equivalence class $[\boldsymbol{u}_n]$, to $\alpha_n \boldsymbol{u}_n$ for some element of the antipodal group $\alpha_n \in \{\pm 1\}$. To obtain a convergence guarantee, we assume that the selection of the signs α_n are made to optimize the chordal L_1 distance to the current mean estimate, that is[1]

$$\alpha_n = \underset{a_n = \pm 1}{\arg\min} \sum_{d=1}^{D} |a_n u_{nd} - \mu_d|. \tag{55}$$

Lemma 3. *Then, with probability* 1, $\mathsf{TGA}(50\%, 1)$ *converges to a local minimum of Eq. 54 in finite time.*

Proof: We shall show that, with probability 1, there exists $M \in \mathbb{N}_0$ such that $\alpha_n = 1$ for all n in every iteration after the M^{th} iteration of the algorithm. Moreover, the value of the energy function

$$\sum_{n=1}^{N} \sum_{d=1}^{D} |u_{nd} - \mu_d| \tag{56}$$

decreases strictly for steps 1 to $(M-1)$, that is

$$\sum_{n=1}^{N} \sum_{d=1}^{D} |\alpha_n u_{nd} - \mu_d| < \sum_{n=1}^{N} \sum_{d=1}^{D} |u_{nd} - \mu_d| \tag{57}$$

for every iteration up to M.

In the i^{th} iteration, with probability 1, we have

$$\sum_{d=1}^{D} |-1 \cdot u_{nd} - \mu_d| \neq \sum_{d=1}^{D} |u_{nd} - \mu_d| \tag{58}$$

for every $n = 1 \ldots N$, because the set on which $\sum_{d=1}^{D} |-1 \cdot u_{nd} - \mu_d| = \sum_{n=1}^{N} \sum_{d=1}^{D} |u_{nd} - \mu_d|$ has measure 0.

Now, we could have $\alpha_n = 1$ for all n, in which case the algorithm has converged and $i \geq M$. Otherwise, there exists some n for which $\alpha_n = -1$ gives

$$\sum_{d=1}^{D} |\alpha_n \cdot u_{nd} - \mu_d| < \sum_{d=1}^{D} |u_{nd} - \mu_d|, \tag{59}$$

in which case the energy in Eq. 56 will decrease strictly in the i^{th} iteration.

The fact that M exists and the TGA algorithm converges in a finite number of steps follows from the fact that there are only finitely many ways to change the sign of $\boldsymbol{u}_{1:N}$, each giving a fixed value of the energy function (56), so there cannot be an infinite sequence of strictly decreasing values.

As a very small perturbation of the data points will not lead to a change in the signs α_n, the algorithm must moreover converge to a local optimum. □

1. In practical implementations we pick α_n to optimize the L_2 distance rather than the L_1 as this can be done highly efficiently.