Iteratively Reweighted Least Squares Algorithm for Sparse Principal Component Analysis with Application to Voting Records

Tomáš Masák [1] | *Charles University, Prague, Czech Republic*

Abstract

Principal component analysis (PCA) is a popular dimensionality reduction and data visualization method. Sparse PCA (SPCA) is its extensively studied and NP-hard-to-solve modifcation. In the past decade, many diferent algorithms were proposed to perform SPCA. We build upon the work of Zou et al. (2006) who recast the SPCA problem into the regression framework and proposed to induce sparsity with the l_1 penalty. Instead, we propose to drop the l_1 penalty and promote sparsity by re-weighting the l_2-norm. Our algorithm thus consists mainly of solving weighted ridge regression problems. We show that the algorithm basically attempts to fnd a solution to a penalized least squares problem with a non-convex penalty that resembles the l_0-norm more closely. We also apply the algorithm to analyze the voting records of the Chamber of Deputies of the Parliament of the Czech Republic. We show not only why the SPCA is more appropriate to analyze this type of data, but we also discuss whether the variable selection property can be utilized as an additional piece of information, for example to create voting calculators automatically.

Keywords

Sparse principal components, IRLS, penalized least squares, roll-calldata

JEL code

C38, C55

INTRODUCTION

Principal component analysis (PCA) is a popular dimensionality reduction technique, which has been successfully applied in virtually all areas of science where multivariate data are encountered. PCA is commonly used in statistics, machine learning, signal processing, genetics, fnance, or meteorology, just to name a few. PCA projects data onto a lower dimensional subspace spanned by the leading eigenvectors of the sample covariance matrix. Sparsity, i.e. restriction on the l_0-norm of the eigenvectors, is often assumed to cope with the issues of poor interpretability and inconsistency of classical PCA in the case of high-dimensional data. Due to the immense growth of data in many disciplines, sparse PCA remains to be area of active research for more than adecade now.

[1] Faculty of Mathematics and Physics, Sokolovská 83, 186 75 Prague 8, Czech Republic. E-mail: tom.masak@gmail.com.

The contribution of our paper is threefold: proposition of a non-convex penalty to be employed in the regression-based PCA setup of Zou et al. (2006), derivation of an iterative algorithm for the resulting non-convex problem together with the proof of its convergence, and application of the new algorithm on a detailed analysis of real-world data. Thus, the paper has three fundamental parts, as outlined in the forthcoming paragraphs.

After a review of (PCA) and the notion of sparsity (Section 1.2), we analyze penalized least squares problem with a non-convex penalty in Section 1.3. We relax the problem introducing additional variables and propose an alternating minimization algorithm to optimize it. The algorithm solves a simple convex optimization problem in every step while the overall algorithm attempts to solve a non-convex problem. We show numerical convergence of the algorithm to a stationary point of the original objective function. Proofs themselves are postponed to the Appendix.

Second, in Section 1.4, we review the S-PCA algorithm of Zou et al. (2006), which possesses several properties that make it attractive in practice, and introduce two of its modifications. First, we describe a very simple modification based on idea of the adaptive lasso (Zou, 2006). This simple modification significantly improves the numerical performance of the S-PCA algorithm, but it also lacks some of the favorable properties of S-PCA. Second, we propose a modification of S-PCA adopting the non-convex penalty function analyzed in Section 1.3. This leads to a new algorithm for sparse PCA. Even though the algorithm is heuristic to some extent, it preserves the favorable properties of the S-PCA algorithm while showing a superior performance.

Finally, we display the differences in performance of the considered algorithms on both real and simulated data in Section 2. We carefully analyze the real data: the voting records of the Chamber of Deputies of the Parliament of the Czech Republic. We examine thoroughly why SPCA is more appropriate than PCA for these data, and what is the value added by the sparsity assumption. The newly proposed algorithm is shown to provide the most easily interpretable results.

The paper is concluded with final remarks in the Discussion.

1 METHODS

1.1 Notation

For a matrix $\mathbf{A} \in \mathbb{R}^{n\times p}$ we denote $\mathbf{a}^{(i)}$ its i-th row, $\mathbf{a}_j$ its j-th column, and a_{ij} its element on the position (i, j). A vector $\mathbf{v} \in \mathbb{R}^p$ consists of entries $v_1, \ldots, v_p$. We also write $\mathbf{v} = [v_i]_{i=1}^p$. All vectors are column-wise, and $\mathbf{I}_k$ stands for the $k \times k$ identity matrix.

For a function $f(\mathbf{x}, \mathbf{y})$ of two vector variables $\mathbf{x} \in \mathbb{R}^p$, $\mathbf{y} \in \mathbb{R}^q$, we denote $\nabla_{\boldsymbol{x}} f(\boldsymbol{x}, \boldsymbol{y})$ the vector of partial derivatives $[\frac{\partial}{\partial x_i} f(\boldsymbol{x}, \boldsymbol{y})]_{i=1}^p$. We abuse the notation slightly when by $\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_0, \boldsymbol{y}_0)$ we denote the part of the gradient evaluated in $(\boldsymbol{x}_0, \boldsymbol{y}_0) \in \mathbb{R}^{p\times q}$.

Finally, note that we abbreviate "sparse principal component analysis" as SPCA while S-PCA refers to a specific algorithm.

1.2 Principal component analysis and sparsity

Suppose we work with a (column-wise) centered data matrix $\mathbf{X} \in \mathbb{R}^{n\times p}$ with the p-dimen-sional observations $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n)}$ in its rows. The objective of PCA is to find linear combinations of the original variables (columns of $\mathbf{X}$) that are uncorrelated and subsequently explain as much variance in the data as possible. Keeping only the first few combinations corresponds to the projection of the original data onto a lower dimensional subspace (called *principal subspace*) best approximating the data in the sum-of-squares sense. One of many equivalent formulations how to find the K-dimensional principal subspace is

$$\underset{\mathbf{X}_\star \in \mathbb{R}^{\mathbf{n}\times\mathbf{p}}}{\arg\min} \quad \|\mathbf{X} - \mathbf{X}_\star\|_F^2 \quad \text{s.t.} \quad \text{rank}(\mathbf{X}_\star) = K, \tag{1}$$

where $\|\cdot\|_F$ is the Frobenius norm, i.e for $\mathbf{A} \in \mathbb{R}^{n\times p}$ it is $\|\mathbf{A}\|_F = \left(\sum_{i=1}^n \sum_{j=1}^p a_{ij}^2\right)^{\frac{1}{2}}$.

Although this problem is non-convex due to the rank constraint, it can be solved explicitly via the singular value decomposition (SVD) of the data matrix $\mathbf{X}$, i.e. $\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^\top = \sum_{i=1}^m d_i \mathbf{u}_i \mathbf{v}_i^\top$, where $m = \min(n, p)$, $\mathbf{U} \in \mathbb{R}^{n\times m}$ and $\mathbf{V} \in \mathbb{R}^{p\times m}$ are orthonormal matrices, and $\mathbf{D} \in \mathbb{R}^{m\times m}$ is a diagonal matrix with non-increasing entries $d_1 \geq \dots d_m \geq 0$ on the diagonal. It follows from the Eckart-Young theorem (Eckart and Young, 1936) that a solution to (1) is obtained by truncating the SVD of $\mathbf{X}$, i.e. $\widehat{\mathbf{X}}_\star = \sum_{i=1}^K d_i \mathbf{u}_i \mathbf{v}_i^\top$.

Recall that $\mathbf{v}_1, \dots, \mathbf{v}_K$ are called *loadings*, and they form an orthonormal basis of the principal subspace. They also correspond to the directions capturing subsequently the majority of variance in the data, or equivalently, they are eigenvectors of the sample covariance matrix $\widehat{\mathbf{\Sigma}} = \frac{1}{n}\mathbf{X}^\top\mathbf{X}$. Thus $\mathbf{v}_1, \dots, \mathbf{v}_K$ are estimators of the *population loadings* – the eigenvectors of the population covariance matrix $\mathbf{\Sigma}$. Vectors $\mathbf{u}_1, \dots, \mathbf{u}_K$ are projections of the data onto the directions of loadings and are usually called *principal components* (PCs).

Suppose we are interested in the first K principal components. Problem (1) can be rewritten in terms of finding an orthonormal basis $\mathbf{V}$ of the principal subspace as

$$\underset{\mathbf{V}_\star \in \mathbb{R}^{p\times K}}{\arg\min} \quad \sum_{i=1}^n \|\mathbf{x}^{(i)} - \mathbf{V}_\star\mathbf{V}_\star^\top\mathbf{x}^{(i)}\|_2^2 \quad \text{s.t.} \quad \mathbf{V}_\star^\top\mathbf{V}_\star = \mathbf{I}_K. \tag{2}$$

This equivalence follows by rewriting $\sum_{i=1}^n \|\mathbf{x}^{(i)} - \mathbf{V}_\star\mathbf{V}_\star^\top\mathbf{x}^{(i)}\|_2^2 = \|\mathbf{X} - \mathbf{X}\mathbf{V}_\star\mathbf{V}_\star^\top\|_F^2$ and considering the SVD of $\mathbf{X}$. Understanding PCA from the perspective of finding a basis of the lower-dimensional subspace best fitting the data points $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)} \in \mathbb{R}^p$ will become handy in the upcoming sections.

PCA is also connected to *factor analysis*, even though the objectives of PCA and factor analysis often differ (see Jolliffe, 2002, Section 7.3, for a discussion). Suppose that data $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}$ are generated from the following factor model:

$$\mathbf{x} = \mathbf{W}\mathbf{f} + \boldsymbol{\epsilon}, \tag{3}$$

where $\mathbf{W} \in \mathbb{R}^{p\times K}$ is a fixed unknown matrix, $\mathbf{f} \in \mathbb{R}^K$ is a random vector (of so-called *factors*), and $\boldsymbol{\epsilon} \in \mathbb{R}^p$ is a random error term, such that $\mathbb{E}\mathbf{f} = \mathbf{0} = \mathbb{E}\boldsymbol{\epsilon}$, $\boldsymbol{\epsilon}$ and $\mathbf{f}$ are independent, $\text{Var}(\mathbf{f}) = \mathbf{I}_K$, and $\text{Var}(\boldsymbol{\epsilon}) = \sigma^2\mathbf{I}_p$ for some $\sigma^2 > 0$. Under this model, it holds $\mathbf{\Sigma} = \text{Var}(\mathbf{x}) = \mathbf{W}\mathbf{W}^\top + \sigma^2\mathbf{I}_p$. Hence, population loadings form an orthonormal basis of the column space of $\mathbf{W}$. Thus PCA can be regarded as an attempt to estimate an orthonormal basis of the subspace spanned by $\mathbf{W}$, i.e. the subspace to which the factors are projected. In other words, PCA can be loosely thought of as describing some hidden factors in the domain of the observed variables.

PCA works well and the theory behind it is well understood in the traditional setting with a fixed, low number of variables p and a higher number of observations n. Nonetheless, PCA encounters both theoretical and practical difficulties when p is comparable to or larger than the sample size n. At various levels of generality, a set of papers showed that eigenvectors of the sample covariance matrix are not consistent estimators of their population counterparts in the joint limit case $p, n \to \infty$, $p/n \to \tau \geq 0$ (e.g. Paul, 2007, Nadler, 2008, and references therein). Essentially, in the high-dimensional case when $p > n$, it is hopeless for loadings to be credible estimates of their population

counterparts. Moreover, the interpretability of PCA is problematic even for a moderate p because all the coefficients of the loadings are nonzero.

To regain the lost consistency of PCA in the high-dimensional setup, an additional structure needs to be assumed. The most widely adopted structure assumption is *sparsity* of the loadings $\mathbf{v}_1, \ldots, \mathbf{v}_K$. A vector $\mathbf{v}$ is s-sparse if $\|\mathbf{v}\|_0 \leq s$, where $\|\mathbf{v}\|_0 = \{\#i; v_i \neq 0\}$ is the *cardinality* of $\mathbf{v}$, i.e. the number of nonzero coefficients of $\mathbf{v}$ or the so-called ℓ_0-norm.[1] Under the sparsity assumption, $\mathbf{v}_i$ is presumed to be s_i-sparse for some $s_i < p$. If the values s_i are sufficiently small, it is possible to estimate the loadings consistently even in the high-dimensional case. Sparsity also greatly improves the interpretability of PCA.

However, while standard PCA is an easy task equivalent to SVD of the data matrix, sparse PCA (SPCA) constraining the numbers of nonzero loadings coefficients is known to be an NP-hard problem. SPCA has been a topic of an active research for more then a decade, and many different algorithms for SPCA have been developed. We build upon the procedure proposed by Zou et al. (2006), which enjoys several favorable properties, especially from the practical point of view. Before introducing this algorithm, we discuss a non-convex penalized least squares problem to be utilized later.

1.3 Non-convex sum-of-sguares penalization

The following penalized least squares problem is extensively studied in the literature:

$$\underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\arg\min} \quad \|\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda\rho(\boldsymbol{\beta}), \tag{4}$$

where $\lambda > 0$ is a tuning parameter, $\mathbf{Y} \in \mathbb{R}^n$, $\mathbf{X} \in \mathbb{R}^{n\times p}$, and $\rho : \mathbb{R}^p \to \mathbb{R}$ is a penalty function. The most prevalent choices of the penalty function are $\rho(\cdot) \equiv \|\cdot\|_2$, which leads to the *ridge regression* (Hoerl and Kennard, 1970), or $\rho(\cdot) \equiv \|\cdot\|_1$, which leads to the *lasso* (Tibshirani, 1996). Optimally, we would like to solve the problem with $\rho(\cdot) \equiv \|\cdot\|_0$ but this leads to an NP-hard problem. The lasso is often used as a convex relaxation to the ℓ_0-norm.

In this paper, we work with the following penalty:

$$\rho_l(\boldsymbol{\beta}|\delta) = \sum_{j=1}^{p} \log(\beta_j^2 + \delta), \tag{5}$$

where $\delta > 0$ is a regularizing parameter, because it resembles more closely the ℓ_0-norm as shown in Figure 1 and discussed in Example 1. However, the problem

$$\underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\arg\min} \quad f(\boldsymbol{\beta}|\delta) \quad \equiv \quad \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\arg\min} \quad \|\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda \sum_{j=1}^{p} \log(\beta_j^2 + \delta). \tag{6}$$

is still non-convex and thus hard to solve. Therefore, we introduce auxiliary variables and propose an alternating minimization algorithm enabling to solve the relaxed problem.

We define a surrogate function

$$g(\boldsymbol{\beta}, \boldsymbol{w}|\delta) = \|\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda \sum_{j=1}^{p} w_j\beta_j^2 + \lambda \sum_{j=1}^{p} \big[w_j\delta - \log(w_j) - 1\big], \tag{7}$$

[1] We note here that $\|\cdot\|_0$ is actually not a norm, but the term is traditional due to Donoho (2006).

where $w_j > 0$ for $j = 1, \dots, p$, and propose to solve (6) by minimizing g over one of the variables $\boldsymbol{\beta}$ and $\mathbf{w}$, keeping the other variable fixed. Starting from some $\mathbf{w}^{(0)}$ (for example $w_j^{(0)} = 1$, $j = 1, \dots, p$), this leads to the following iterative scheme:

$$\boldsymbol{\beta}^{(l+1)} = \underset{\boldsymbol{\beta} \in \mathbb{R}^p}{\arg\min} \; g(\boldsymbol{\beta}, \boldsymbol{w}^{(l)} | \delta) \tag{8}$$

$$\boldsymbol{w}^{(l+1)} = \underset{\boldsymbol{\beta} \in \mathbb{R}^p}{\arg\min} \; g(\boldsymbol{\beta}^{(l+1)}, \boldsymbol{w} | \delta) \tag{9}$$

Lemma 1. For f of (6) and its surrogate g of (7), the following properties hold:

(a) $\forall \delta > 0, \forall \boldsymbol{\beta} \in \mathbb{R}^p$:

$$f(\boldsymbol{\beta}|\delta) = \min_{\boldsymbol{w} \in \mathbb{R}^p} \; g(\boldsymbol{\beta}, \boldsymbol{w} | \delta).$$

(b) $\forall l = 1, 2, \dots \forall \delta > 0$: $\boldsymbol{\beta}^{(l+1)}$ of (8) is the unique solution to the weighted ridge regression problem

$$\underset{\boldsymbol{\beta} \in \mathbb{R}^p}{\arg\min} \; \|\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda \sum_{j=1}^{p} w_j^{(l)} \beta_j^2.$$

(c) $\forall l = 1, 2, \dots \forall \delta > 0$: $\boldsymbol{w}^{(l+1)}$ of (9) is given by

$$w_j^{(l+1)} = \tfrac{1}{(\beta_j^{(l+1)})^2 + \delta}, \quad j = 1, \dots, p. \tag{10}$$

The proof consists of simple calculations, and hence we omit it.

Example 1. Let $\mathbf{X} \in \mathbb{R}^{n \times 2}$, $n \in \mathbb{N}$, $\mathbf{Y} \in \mathbb{R}^n$ and suppose $\mathbf{Y} \approx \mathbf{X}\boldsymbol{\beta}$ for an unknown $\boldsymbol{\beta} \in \mathbb{R}^2$. Thus we are in a situation with two regressors. Suppose we want to find an estimate of $\boldsymbol{\beta}$ as a minizimizer of the sum of squares of residuals but having cardinality at most one, i.e. we wish to solve

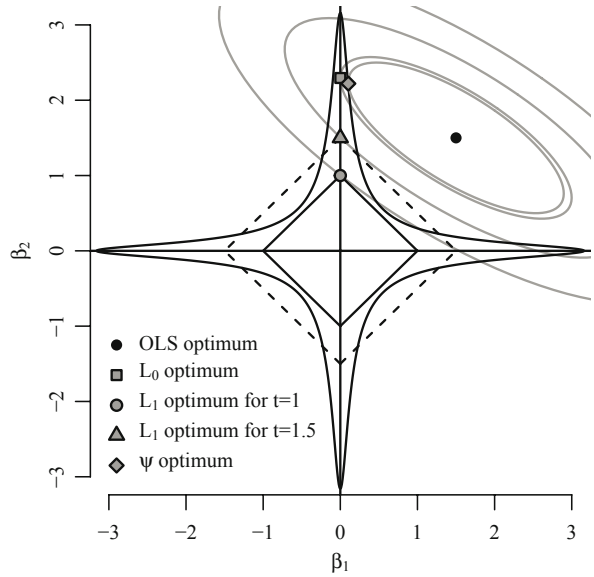
$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta} \in \mathbb{R}^2}{\arg\min} \; \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 \quad \text{s.t.} \quad \rho(\boldsymbol{\beta}) \leq t, \tag{11}$$

for $\rho(\cdot) \equiv \|\cdot\|_0$ and $t = 1$. Note that the permissible set $\{\boldsymbol{\beta} \in \mathbb{R}^2, \|\boldsymbol{\beta}\|_0 \leq 1\}$ corresponds to the whole coordinate axes. Thus, in Figure 1, the estimate corresponds to the point of contact of the coordinate axes with the ellipse (representing a sum-of-squares level set) closest to the ordinary least squares (OLS) estimate. In Figure 1, also solutions to (11) for $\rho(\cdot) \equiv \|\cdot\|_1$ (i.e. lasso estimates) with $t = 1$ and $t = 1.5$ are given. Note that the lasso estimate for $t = 1.5$ is the closest ℓ_1-penalized approximation to the ℓ_0-penalized solution one can get. Increasing t any further results in losing the sparsity and the lasso estimate actually grows away from the desired point closer to the OLS estimate. On the other hand, the solution to (11) for $\rho(\cdot) \equiv \rho_l(\cdot|\delta)$ with appropriate t and δ is able to approximate the ℓ_0 solution to arbitrary precision. In figure 1, the solution with $\delta = 10^{-2}$ and $t = \frac{1}{2}\log(\delta)$ is shown. Note that for a fixed t, it holds for $\delta_1 > \delta_2$ that

$$\Big\{\boldsymbol{\beta} \in \mathbb{R}^p; \sum_{j=1}^{p} \log(\beta_j^2 + \delta_1) \leq t\Big\} \subset \Big\{\boldsymbol{\beta} \in \mathbb{R}^p; \sum_{j=1}^{p} \log(\beta_j^2 + \delta_2) \leq t\Big\},$$

thus it is necessary to vary both t and δ to approximate the ℓ_0-norm well. This translates to the necessity to both decrease δ and increase λ accordingly in the unconstrained formulation of the problem (6).

Figure 1 Geometry of the ℓ_0, ℓ_1, and p_l penalties in the case of two regressors. Solutions to the constrained problem (11) with various penalties along with the unpenalized OLS solution are shown



Source: Own construction

Lemma 1 (a) shows that successful minimization of g results in the minimization of f. It follows from Lemma 1 (b) that the alternating minimization scheme of (8) and (9) consists of a sequence of weighted ridge regression problems, thus it is in fact an instance of iteratively reweighted least squares (IRLS) algorithm. From Lemma 1 (a) it also follows the algorithm is a majorization-minimization (MM) algorithm (Lange et al., 2000).

Lemma 2. Let $w_j^{(0)} > 0$, $j = 1, \ldots, p$, let $\delta \geq 0$, and let $\{\boldsymbol{\beta}^{(l)}\}_{l=1}^{\infty}$ be the sequence obtained by solving (8) and (9) for $l = 1, 2, \ldots$. Then $\|\boldsymbol{\beta}^{(l+1)} - \boldsymbol{\beta}^{(l)}\|_2 \to 0$ and every limit point of the sequence $\{\boldsymbol{\beta}^{(l)}\}_{l=1}^{\infty}$ is a stationary point of $f(\boldsymbol{\beta}|\delta)$.

The previous lemma can be proved easily by following the general theory of MM algorithms (Lange et al., 2000). We do not show this here since the lemma also follows from more general Proposition 3, which we prove in detail. Nonetheless, Lemma 2 implies numerical convergence of the iterative scheme consisting of (8) and (9). It also implies that the algorithm effectively tries to minimize $f(\boldsymbol{\beta}|\delta)$, although convergence to the true global minimum can not be guaranteed for the non-convex problem.

Remark 2. Weighted ridge regression can be transformed into regular ridge regression by a simple transformation of data. Furthermore, a ridge regression problem can be efficiently solved via the SVD of the data matrix. Thus the iterative scheme (8)-(9) operates by repeatedly executing SVD.

Performance studies of similar algorithms strongly suggest that δ should not stay fixed throughout the iterations (Daubechies et al., 2010). We would prefer to have δ very small to approximate ℓ_0-norm well with the penalty function ρ_l, but the smaller δ the more non-convex problem (6) gets. Subsequently, the greater is the chance the algorithm gets trapped in local minima near the starting point. Thus we would like to start with a relatively large δ to avoid local minima near the

Algorithm 1 Interative algorithm for problem (6) with varying δ

Input: $w_j^{(0)} \geq 0$ (default: $w_j^{(0)} =$), $j = 1, \ldots, p$, $\xi > 0$,
a sequence $\{\delta_l\}_{l=0}^{\infty}$ such that $\delta_0 \geq \delta_1 \geq \ldots$ and $\delta_l \to \delta > 0$ (default: $\delta_0 = 1$ and $\delta_{l+1} = \max(\delta_l / 1.5, \delta_{\min})$, where $\delta_{\min} = 10^{-6}$).

Output: Estimate $\widehat{\boldsymbol{\beta}}$.

begin
  Set $l = 0$.
  repeat
    $$\boldsymbol{\beta}^{(l+1)} = \underset{\boldsymbol{\beta} \in \mathbb{R}^p}{\arg\min} \; g(\boldsymbol{\beta}, \boldsymbol{w}^{(l)} | \delta_l) \tag{12}$$
    $$\boldsymbol{w}^{(l+1)} = \underset{\boldsymbol{\beta} \in \mathbb{R}^p}{\arg\min} \; g(\boldsymbol{\beta}^{(l+1)}, \boldsymbol{w} | \delta_{l+1}) \tag{13}$$
    Set $l = l + 1$.
  until $\|\boldsymbol{\beta}^{(l)} - \boldsymbol{\beta}^{(l-1)}\|_2 < \xi$
  Set $\widehat{\boldsymbol{\beta}} = \boldsymbol{\beta}^{(l)}$.
end

initialization, and decrease δ gradually during the iterations. This leads to Algorithm 1.

Lemma 1 (b) and (c) still indicate how the solutions of (1) and (1) look like. The only difference is that δ in (10) is replaced by δ_{l+1}. On the other hand, Lemma 1 (a) is useless now. It is not a priori clear which function Algorithm 1 effectively tries to minimize, if any, and Algorithm 1 is no longer an MM algorithm. Despite the fact, we still have numerical convergence to a stationary point of the limit function $f(\boldsymbol{\beta}|\delta)$, where δ is now the limit of the sequence of $\{\delta_l\}_{l=1}^{\infty}$.

Proposition 3. Let $w_j^{(0)} > 0$, $j = 1, \ldots, p$, let $\delta_0 \geq \delta_1 \geq \ldots$ be a sequence such that $\delta_l \to \delta_\star > 0$ and let $\{\boldsymbol{\beta}^{(l)}\}_{l=1}^{\infty}$ be the sequence generated by Algorithm 1. Then $\|\boldsymbol{\beta}^{(l+1)} - \boldsymbol{\beta}^{(l)}\|_2 \to 0$ and every limit point of the sequence $\{\boldsymbol{\beta}^{(l)}\}_{l=1}^{\infty}$ is a stationary point of $f(\boldsymbol{\beta}|\delta_\star)$.

We defer the proof to the Appendix.

1.4 S-PCA algorithm

Zou et al. (2006) proposed a suitable reformulation of the PCA problem. They showed that for any $\mu > 0$ the solution $(\widehat{\mathbf{A}}, \widehat{\mathbf{B}})$ of

$$\underset{\mathbf{A}, \mathbf{B} \in \mathbb{R}^{p \times K}}{\arg\min} \; \sum_{i=1}^{n} \|\mathbf{x}^{(i)} - \mathbf{A}\mathbf{B}^\top \mathbf{x}^{(i)}\|_2^2 + \mu \sum_{k=1}^{K} \|\mathbf{b_j}\|_2^2 \quad \text{s.t.} \quad \mathbf{A}^\top \mathbf{A} = \mathbf{I}_K \tag{14}$$

satisfies that the normalized columns of $\widehat{\mathbf{B}}$ are exactly the first K loadings $\mathbf{v}_1, \ldots, \mathbf{v}_K$ (possibly up to a rotation). We invite the reader to compare (14) to (2).

With the goal to induce sparsity of the loadings, Zou et al. (2006) proposed to incorporate ℓ_1 penalties in (14) leading to

$$\begin{aligned} \underset{\mathbf{A}, \mathbf{B} \in \mathbb{R}^{p \times K}}{\arg\min} \; & \sum_{i=1}^{n} \|\mathbf{x}^{(i)} - \mathbf{A}\mathbf{B}^\top \mathbf{x}^{(i)}\|_2^2 + \mu \sum_{k=1}^{K} \|\mathbf{b}_j\|_2^2 + \sum_{k=1}^{K} \lambda_k \|\mathbf{b}_j\|_1 \\ \text{s.t.} \quad & \mathbf{A}^\top \mathbf{A} = \mathbf{I}_K, \end{aligned} \tag{15}$$

where $\lambda_1, \ldots, \lambda_K > 0$. For a fixed $\mathbf{A}$, (15) can be rewritten as K independent problems of the form

$$\underset{\boldsymbol{\beta} \in \mathbb{R}^p}{\arg\min} \quad \|\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \mu\|\boldsymbol{\beta}\|_2^2 + \lambda\|\boldsymbol{\beta}\|_1, \tag{16}$$

where $\mu, \lambda > 0$, while, for a fixed $\mathbf{B}$, the solution of (15) is simply given by a suitable rotation.

Zou et al. (2006) proposed an alternating algorithm for problem (15), fixing either $\mathbf{A}$ or $\mathbf{B}$ in every step and solving for the other term. The algorithm is called S-PCA.

Unfortunately, problem (15) is still non-convex, and Zou et al. (2006) did not provide any theoretical guarantees for the convergence of their alternating minimization algorithm. Although numerical convergence to some fixed point could be proven easily (but we do not pursue this due to the space restrictions), the algorithm is not guaranteed to provide a global minimum (Witten et al., 2009). On the other hand, empirical results show that ordinary loadings (obtained via SVD) provide a reliable starting point, and the algorithm does its job.

Remark 3. Even though there are no strong theoretical guarantees for the S-PCA algorithm, the algorithm may very well be the method of choice for many applications because it enjoys several favorable properties lacked by its competitors. These are:

(a) an exact cardinality (i.e. the desired number of nonzero coefficients) can be chosen for each vector of loadings separately,
(b) as the cardinality constraints are lifted, the method gradually reduces to classical PCA,
(c) a chosen number K of sparse PCs is computed at once,
(d) the computed loadings are truly orthonormal, and
(e) the method does not require neither storage nor computation of the covariance matrix (which can be unmanageable for $p > n$).

However, the S-PCA algorithm is outdated in terms of performance, as will be shown in section 2. Hence our goal is to develop a modification that would retain the above stated favorable properties and be competitive at the same time. Since the cornerstone of the S-PCA algorithm is repeatedly solving (16) (called *elastic net* problem, see Zou and Hastie, 2005), a natural way to improve the algorithm's performance is to use the large body of literature dealing with penalization in the regression context. We can substitute the elastic net problem (16) for a different one. This adds up to using different penalty terms in (15).

1.4.1 ADA-S-PCA modification

The most natural way to improve performance of the S-PCA algorithm is to use the *adaptive elastic net* (Zou and Zhang, 2009) instead of (16), i.e. to solve

$$\begin{aligned} \underset{\mathbf{A},\mathbf{B} \in \mathbb{R}^{p \times K}}{\arg\min} \quad & \sum_{i=1}^{n} \|\mathbf{x}^{(i)} - \mathbf{A}\mathbf{B}^\top \mathbf{x}^{(i)}\|_2^2 + \mu \sum_{k=1}^{K} \|\mathbf{b_j}\|_2^2 + \sum_{k=1}^{K} \lambda_k \sum_{j=1}^{p} w_{kj}|b_{kj}| \\ \text{s.t.} \quad & \mathbf{A}^\top \mathbf{A} = \mathbf{I}_K \end{aligned} \tag{17}$$

where $\mathbf{w}_k = (w_{k1}, \ldots, w_{kp})^\top$, $k = 1, \ldots, p$, are vectors of weights. The adaptive elastic net allows different degree of penalization for each coefficient. The weights should be small for nonzero coefficients and large for zero ones to balance out the fact that ℓ_1-norm penalizes all the coefficients equally. Thus, the weights w_{kj} should be taken inversely proportional to the true sparse loadings. In that case, mild penalty would be imposed on nonzero coefficients reducing their shrinkage while more severe penalty would be imposed on zero coefficients forcing them to be estimated as zeros.

Since we do not know the true sparse loadings, it is natural to take $w_{kj} = \frac{1}{v_{kj}}$, that is to take the weights proportional to the ordinary loadings obtained from the SVD of $\mathbf{X}$.

One can transform an adaptive elastic net problem to an elastic net problem by a simple transformation of the data. Moreover, the weights are known in our case because the ordinary loadings are the starting point for S-PCA algorithm. Thus the adaptive version (17) (which we refer to as ADA-S-PCA) of the S-PCA algorithm adds up to a very simple modification requiring negligible additional computation time. Still, the adaptive modification noticeably improves the performance of the S-PCA algorithm, which was observed several times (e.g. Chen, 2011, or Leng and Wang, 2012).

On the other hand, property (b) of Remark 3 of the S-PCA algorithm cease to hold for ADA-S-PCA. This implies, since ordinary PCs explain as much variance as possible, that ADA-S-PCA must be highly suboptimal in terms of explained variance in data whenever the desired cardinality is relatively high. We will observe this in Section 2.1.

1.4.2 IRLS-S-PCA modification

We replace the ℓ_1 penalty in (15) by the penalty function ρ_l of (5). Furthermore, the role of the ℓ_2 penalty in (14) is only to ensure uniqueness of the solution. This would be unnecessary for the numerical algorithm we will propose. Thus we drop the ℓ_2 penalty and propose to solve

$$\underset{\mathbf{A},\mathbf{B}\in\mathbb{R}^{p\times K}}{\arg\min} \quad \sum_{i=1}^{n} \|\mathbf{x}^{(i)} - \mathbf{A}\mathbf{B}^\top \mathbf{x}^{(i)}\|_2^2 + \sum_{k=1}^{K} \lambda_k \sum_{j=1}^{p} \log(b_{kj}^2 + \delta) \quad \text{s.t.} \quad \mathbf{A}^\top\mathbf{A} = \mathbf{I}_K \tag{18}$$

with the help of the alternating minimization algorithm of Zou et al. (2006). That is: for a fixed B, update $\mathbf{A} = \mathbf{U}\mathbf{V}^\top$ where $\mathbf{U}$ and $\mathbf{V}$ are from the SVD $\mathbf{X}^\top\mathbf{X}\mathbf{B} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$, and, for a fixed $\mathbf{A}$, solve K independent penalized least squares problems of the form (6) using Algorithm 1. We implemented the IRLS-S-PCA method in software R.[2]

The motivation behind this proposal is the geometry of the penalty function (5). With SPCA we assume that the true (population) loadings are sparse. Since sparsity is defined in terms of ℓ_0-norm, one should penalize ℓ_0-norms in (14) instead of ℓ_1-norms, which would result in best subset selection problems instead of (16). Best subset selection problems are NP-hard due to the ℓ_0 penalty, and ℓ_1 penalty is commonly used as a convex relaxation of the problem. However, non-convex penalties can naturally resemble ℓ_0 penalty better than ℓ_1 penalty does, see Figure 1. This comes at the expense that the problem becomes non-convex, and must be solved iteratively. Nonetheless, we hope that the obtained solution would still be closer to the ℓ_0-penalized optimum compared to the ℓ_1-penalized optimum.

The reason for the concrete choice of the penalty function ρ_l stems from the algorithm we developed to minimize the resulting penalized least squares problem. The algorithm iterates between two easily and analytically solvable tasks. This comes at the expense that the solution obtained via the algorithm is not itself sparse, see again Figure 1. For a practical implementation of the algorithm, one need to ensure sparsity of the obtained loadings by thresholding. Moreover, we want to ensure our algorithm also allows one to pick the target cardinality for every PC. The simplest way is to threshold the desired number of coefficients in the output $\widehat{\boldsymbol{\beta}}$ of Algorithm 1. Instead, we use a cumulative version of the algorithm, in which the weights actually used in (1) are products of the weights computed from (1) in the last few steps, and we iterate until a prescribed number of coefficients decrease under a fixed threshold.

[2] <http:// www.karlin.mff.cuni.cz/masak/irls_s_pca.R>.

Our IRLS-S-PCA *algorithm possesses the same favorable properties as the* S-PCA *algorithm that were discussed in Remark 3*. We ensure the property (a) by thresholding. Property (b) follows from the fact that with no sparsity constraints the method only attempts to find a solution to (14). But the starting point is exactly this solution (the same as in the case of S-PCA) due to the initialization of the weights. And properties (c)-(e) are inherited from the S-PCA algorithm. Moreover, the algorithm shows a superior numerical performance as displayed in Section 2.

1.5 Connections to previous work

Motivated by the search for an estimating procedure, that would be asymptotically as efficient as the oracle procedure knowing the true support in advance (the famous lasso does not have this *oracle property*), Fan and Li (2001) proposed to work with a concave penalty function. They presented an iterative algorithm using local quadratic approximations to obtain the non-convex estimate (so-called SCAD).

Later, Zou and Li (2008) proposed to optimize another concave penalty function via local linear approximations. These are tight enough so that a one-step estimate can be asymptotically as efficient as a fully iterated one, provided a reliable starting point. The well-known *adaptive lasso* (Zou, 2006) is such a one-step estimate. It has the oracle property while consisting of just two convex optimization tasks, and it is the motivation behind the ADA-S-PCA algorithm presented later.

Our penalty function ρ_l does not fall into the framework of the previously mentioned papers directly, because ρ_l is not concave on $(0, \infty)$. This has two consequences. First, we can not use local linear approximations. Algorithm 1, only with $\delta > 0$ fixed, essentially uses local quadratic approximations, and these are not very tight. Thus a fully iterated estimate is needed. On the other hand, these iterations have much simpler form than local linear approximations. Second, our penalty function is differentiable at zero which implies the resulting estimate is not itself sparse. The differentiability also implies that Algorithm 1 with a fixed $\delta > 0$ is an instance of the Newton-Rhapson algorithm.

Another distinction of our approach is that we let the regularizing parameter δ to vary. The regularizing parameter was introduced by Hunter and Li (2005) to relieve the drawback of the backward variable selection of the SCAD estimator of Fan and Li (2001). The varying regularizing parameter does a similar job, and allows us in addition to avoid local minima near the initialization point at the same time (see Daubechies et al., 2010, and references therein).

2 EXAMPLES

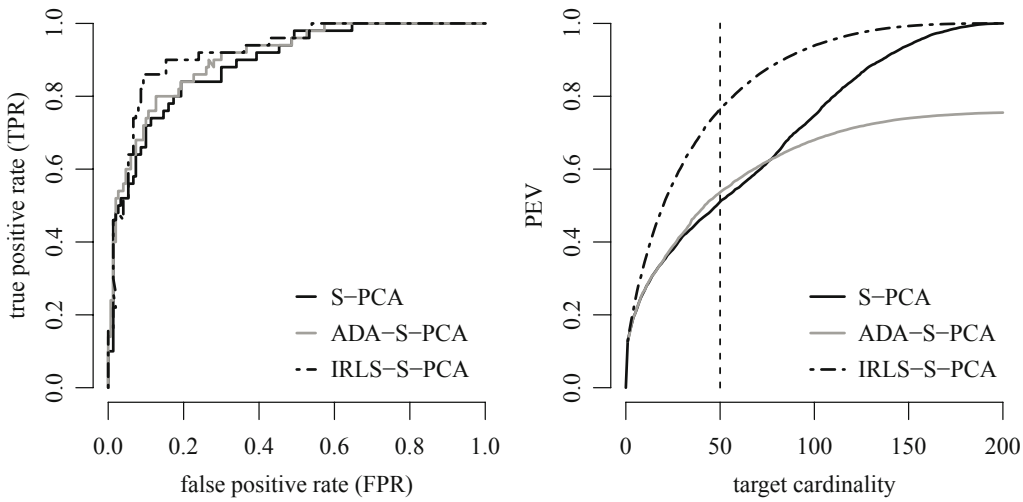
In this section, we first show a simulated example that displays differences between the approaches described above. Then we move to a practical application of SPCA: analysis of voting records of the Chamber of Deputies of the Parliament of the Czech Republic.

2.1 Simulated example

We generate a vector $\widetilde{\mathbf{v}} \in \mathbb{R}^{200}$ with coefficient that are independent, uniformly distributed on the interval (0,1). Then we randomly pick 150 of its coefficients and set them to zero, thus obtaining a 50-sparse vector. We normalize this vector to obtain a 50-sparse unit vector $\mathbf{v}$. Using the procedure described by Shen and Huang (2008), we create a covariance matrix $\boldsymbol{\Sigma}$ with the spectral decomposition

$$\boldsymbol{\Sigma} = \theta \mathbf{v}\mathbf{v}^\top + \sum_{j=2}^{200} \mathbf{v}_j \mathbf{v}_j^\top \in \mathbb{R}^{200\times 200}, \tag{19}$$

Figure 2 *Left:* Sample ROC curves for $\theta = 5$. Higher curve represents better performance. *Right:* Average (over 100 simulation runs) area under the curve as a function of θ.



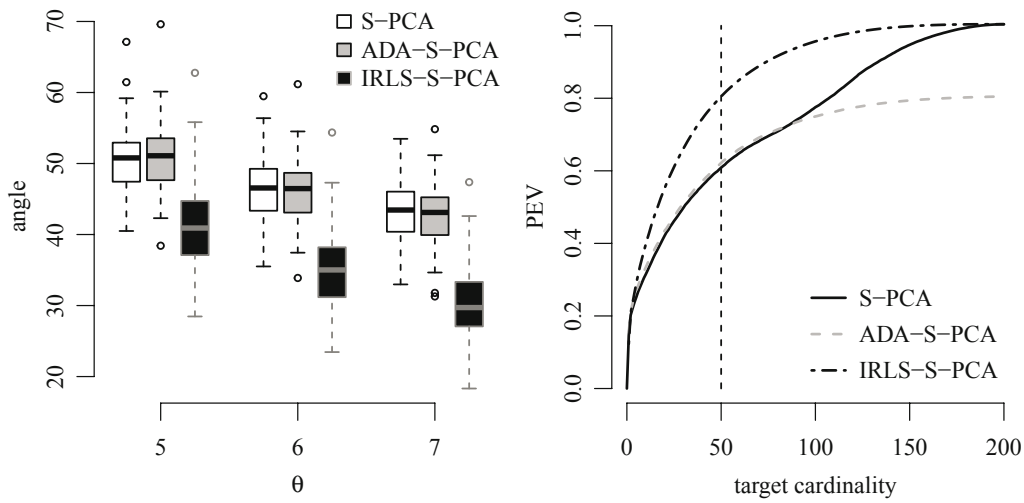
Source: Own construction

where $\theta > 1$, and $\mathbf{v}_2, \ldots, \mathbf{v}_{200} \in \mathbb{R}^{200}$ are orthonormal, all orthogonal to $\mathbf{v}$. Thus the matrix $\mathbf{\Sigma}$ is regular, its leading eigenvector $\mathbf{v}$ is associated with the largest eigenvalue θ, and all the other eigenvalues are equal to one. We generate $n = 100$ independent samples from $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$ to obtain the data matrix $\mathbf{X}$. This procedure is repeated 100 times for every value $\theta \in \{3, 4, 5, 6, 7\}$.

Model (19) was introduced by Johnstone (2001). Later, Paul (2007) and Johnstone and Lu (2009) showed that PCA is not consistent under this particular model in the joint limit setting ($p, n \to \infty$ and $p/n \to \tau$) unless $\tau = 0$. The model is called *spiked covariance* as data sampled from this model are "spiked" in the direction of $\mathbf{v}$ and evenly spread in other directions (representing noise). Vector $\mathbf{v}$ can be thought of as a signal and θ as the signal strength. The stronger the signal is (i.e. θ is larger), more easy it is to distinguish it from noise. We are interested in estimating the signal $\mathbf{v}$ from the data, i.e. we are interested in the first principal component.

First, we are concerned with the quality of the considered algorithms with respect to the estimation of the support of $\mathbf{v}$. For a fixed value of s, every considered algorithm identifies exactly s coefficients as nonzero. It is natural to ask whether the correct coefficients were identified. One can compute (for a fixed s) the *true positive rate* (TPR, the proportion of correctly identified nonzero coefficients) and the *false positive rate* (FPR, the proportion of coefficients that were estimated as nonzeros even though they are truly zero). If we plot these pairs for all the values of s, we obtain an analogy to the so-called ROC curve as shown in the left panel of Figure 2. Note that our employment of the ROC curve differs from the typical usage in the context of binary classifiers. But still, higher curve represents a better performance. Thus it is natural to use the area under the curve as a measure of performance in order to have a single number comparing performances of different algorithms. In the right panel of Figure 2, area under the curve is plotted against the signal strength θ. For every $\theta \in \{3, 4, 5, 6, 7\}$, 100 datasets were simulated by the above-described procedure, and the average values are shown. As we can see in the figure, IRLS-S-PCA and ADA-S-PCA both overcome S-PCA in terms of support estimation while there is only a negligible difference between them.

Figure 3 *Left:* Angles between vector **v** and its estimates from our three methods for $\theta = 5, 6, 7$ with $s = 50$. *Right:* Proportion of explained variance as a function of target cardinality s for $\theta = 5$.



Source: Own construction

Second, correct support estimation is not sufficient for reliable estimation of the respective PC. Thus we compute angles between **v** and its estimates obtained by the considered algorithms with the value of target cardinality s set to 50. Results are plotted in the left panel of Figure 3. Note that the knowledge that **v** is 50-sparse is used solely for the purpose of this figure. Even though the performance of ADA-S-PCA and IRLS-S-PCA with respect to support estimation is comparable, IRLS-S-PCA clearly outperforms ADA-S-PCA regarding the angles between the estimated and true vectors of loadings.

Finally, we are also interested in the amount of explained variance. Since no vector can explain more variance in data than the first classical PC, we plot in the right panel of Figure 3 explained variance for different values of target cardinality s as a proportion of the variance explained by classical PCA.[3] We see that IRLS-S-PCA explains higher proportions of variance than S-PCA, and the proportion of variance for both of these methods approach one as s approaches to 200. This is an empirical evidence that both IRLS-S-PCA and ADA-S-PCA reduce to classical PCA as the cardinality constraint is lifted. On the other hand, ADA-S-PCA doesn't have this property and thus it is suboptimal in terms of explained variance.

2.2 Real-world example: analysis of voting records

2.2.1 Data description and objectives

In this section, we show an application of SPCA on a real data set, namely voting records (so-called *roll-call* data) of the Chamber of Deputies of the Parliament of the Czech Republic. The data are

[3] Again, only average values over 100 simulation runs are shown. This is for the sake of clarity and due to the space restrictions. For the right-panel of Figure 2 the standard deviations are of order 10^{-2} with differences of order 10^{-3} across the respective algorithms. For the right-hand panel of Figure 3 the situation is similar, with the exception that standard deviations of IRLS-S-PCA are one order lower than those of other methods.

publicly available on the official website of the Chamber of Deputies.[4] They give information about deputies, their memberships in political parties and different committees, and most importantly their voting records. The data are organized in several tables and date back to year 1993. We are interested mostly in the table "hl_poslanec" with three variables: ID of the deputy, ID of the act, and the individual vote. The only additional information we use comes from the table "poslanec", which enables us to align the deputies to the parties of their membership for the purposes of our outputs. More detailed description of the data set can be found on the website of the Chamber of Deputies.

Note that, from all available data, we are interested only in the voting records of the year 2015, i.e. a one-year period of the current administration.

We have transformed the raw data (table "hl_poslanec") into a data matrix $\mathbf{X}$ with entries

$$x_{ij} = \begin{cases} 1, & \text{if the } i\text{-th deputy voted "yes" in the } j\text{-th act,} \\ -1, & \text{if the } i\text{-th deputy voted "no" in the } j\text{-th act,} \\ 0, & \text{otherwise.} \end{cases}$$

While the original data distinguish several scenarios, why a given deputy did not vote neither "yes" nor "no" in the given act (e.g. he was registered and abstained from voting, he was registered but failed to vote, or he wasn't present), we regard all those scenarios the same as zeros. This is, of course, mainly due to simplicity but also due to the fact that deputies are well aware that the reason of their abstention is irrelevant. Just for information, there are approximately 35% of zeros in the matrix $\mathbf{X}$. Of course, there are no missing data in the sense that a deputy voted and we do not have the information how.

The political situation in the Chamber of Deputies is following. Since the elections in 2014, there is a majority coalition composed of *CSSD*, *ANO*, and *KDU-CSL*. The opposition is formed by *ODS*, *TOP 09*, *Usvit*, and *KSCM*. As for the political positions of the parties, *ODS* is a traditional right-wing party opposed by the communists of *KSCM* on the left-wing of the spectrum. *TOP 09* is centre-right, *CSSD* is centre-left, *ANO* is centre to centre-right and, finally, *KDU-CSL* holds a centre position.

In 2015, 1837 acts took place, and there were no replacements among the deputies (which are 200 in the Chamber of Deputies) during that year. Thus the matrix $\mathbf{X}$ has 200 rows and 1837 columns. The voting record for the i-th deputy, $i = 1, \ldots, 200$, is represented by the vector $\mathbf{x}^{(i)} \in \{-1, 0, 1\}^{1837}$. One deputy thus correspond to a point in a 1837-dimensional space. From now on, let $n = 200$ and $p = 1837$, i.e. $\mathbf{X} \in \mathbb{R}^{n \times p}$.

PCA regards to finding a lower-dimensional subspace best approximating the 1837-dimen-sional data points. If we take the information about the political affiliation of the representatives into account, an interesting underlying structure in the data could be possibly revealed in the lower dimension. Questions such as

- which parties are homogeneous and which are not,
- which deputies are outliers with respect to their parties, and
- which parties stick together, and which do not share the same view,

would be hopefully answered.

[4] <http://www.psp.cz/sqw/hp.sqw?k=1300>.

2.2.2 Assumptions and their justification

For classical PCA, we need to choose a number of PCs to work with. For SPCA, we further need to choose values of target cardinality for the PCs. To justify our choices, recall a data-generative factor model (3).

First, how many PCs should we take into account? Two is the most prevalent option due to the fact that two-dimensional approximation is most easily comprehensible. From the perspective of model (3), dealing with just two PCs means that voting behavior of every respective deputy can be described as a linear combination of two factors, up to a deputy-specific error. This seems appropriate, as one can expect from the very start that one PC would discriminate between coalition and opposition and the other between the right-wing and left-wing parties.

Second, what values of target cardinality should be chosen? Considering model (3) with $K = 2$, there are two unobserved factors, and we observe noisy realizations of those factors in the domain of acts. The question thus translates to how many acts do we need in order to be able to describe the factors reliably. Note that the sparsity assumption seems very reasonable from this perspective. We have chosen 20 acts to describe the first PC and 15 acts to describe the second PC, that is 35 in total. This concrete choice is based on data-non-related considerations. The lists of important acts created by political scientists or journalists often count between 20 and 35 acts per year. We have simply chosen the sum of the desired target cardinalities to be an upper bound to this expert choice because we also believe that 35 acts should be enough to describe the two PCs.

2.2.3 Results

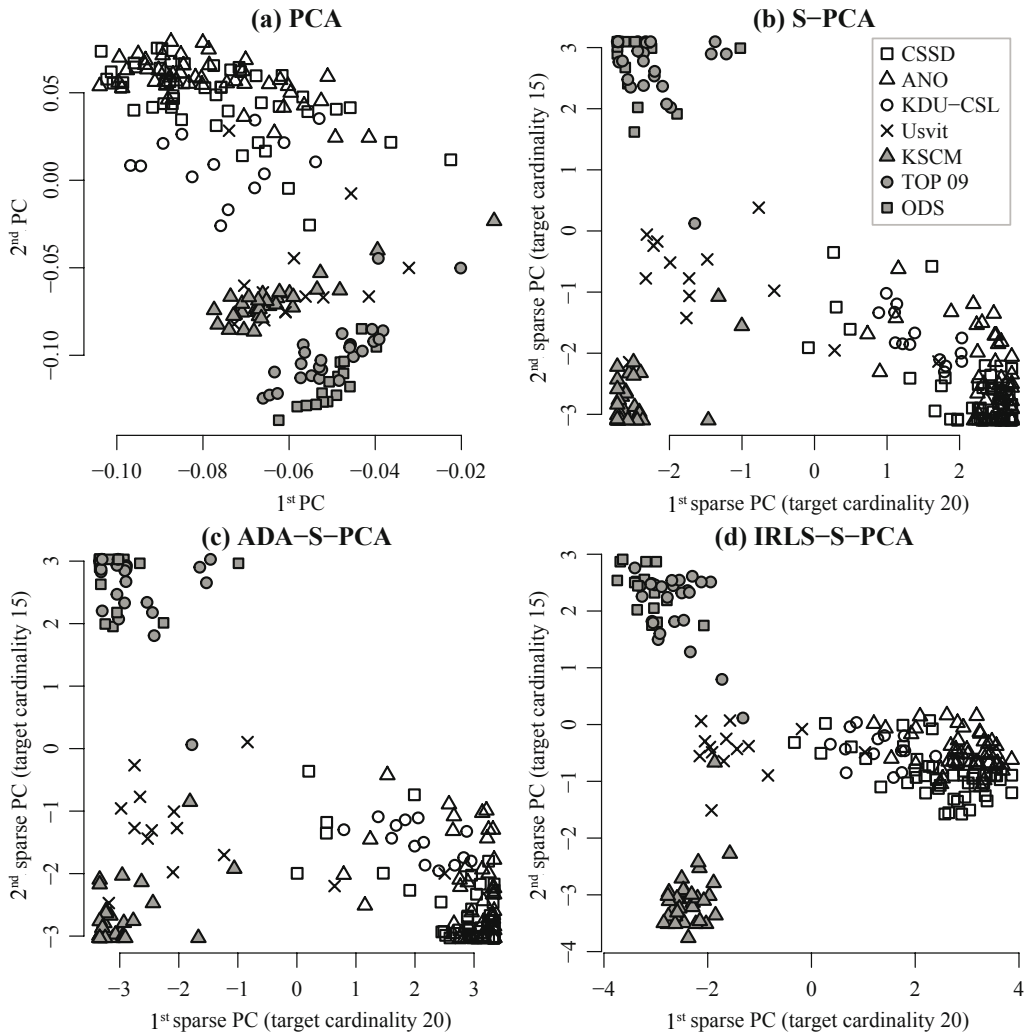
Figure 4 shows results of classical PCA (panel (a)) and SPCA performed by the algorithms presented in Section 1.4 (panels (b)-(d)). All the panels give qualitatively similar results. In all cases, the first sparse PC discriminates between the coalition and the opposition, and the second sparse PC distinguishes between the right and left wings. Even more delicate distinctions can be made. For example, the party *Usvit* is somehow in the middle of the plots, and *KDU-CSL* is the most centered of the coalition parties. Also, one can notice two outliers of *Usvit* in Figure 4 (b): the right-most deputy of the *Usvit* located in between the deputies of *KDU-CSL* and the bottom-left-most deputy surrounded by representatives of *KSCM*. These two outliers can be observed also in (c) and (d) of Figure 4, and they are truly the same deputies.

Nonetheless, the IRLS-S-PCA *algorithm* (Figure 4 (d)) *gives much smoother and more easily interpretable results than the other algorithms*. It sharply discriminates between the coalition/opposition and left-wing/right-wing features, thus the coalition is shown to be in the centre political position. Also, *ANO* is better separated from *CSSD* referring more closely to their expected position. To sum up, Figure 4 (d) captures best the general perception of the Czech political scene.

SPCA reveals a useful additional piece information as opposed to methods commonly used for analysis of voting data (see Clinton et al., 2004, for an overview) that do not assume sparsity. Namely, the variable selection property indicate which acts are important. Sadly, even though panels (b)-(d) of Figure 4 are qualitatively similar, the acts selected to describe the sparse PCs differ noticeably among the methods. There are only three acts that were selected by all methods. These are:

- act no. 60 403, on the agenda of the meeting, specifically whether a specific amendment should be considered first,
- act no. 61 973, on an amendment on the state budget draft for 2016,
- act no. 62 034, on the state budget draft as a whole.

Figure 4 Voting records of the Chamber of Deputies of the Parliament of the Czech Republic projected onto a plane given by: (a) first two ordinary PCs, (b) first two sparse PCs obtained via S-PCA, (c) first two sparse PCs obtained via ADA-S-PCA, (d) first two sparse PCs obtained via IRLS-S-PCA. For (b)-(d) the cardinality is 20 for the first sparse PC and 15 for the second sparse PC.



Source: Own construction

On one hand, it is convenient that the act on the state budget draft as a whole was picked by all the methods. On the other hand, it is unseemly that behind the act no. 60 403 there is a hidden agenda, and it takes an expert knowledge and going through the stenographic records to discover what the agenda is. The same holds for the act no. 61 973. Nonetheless, this does not change the fact that the sparse representation of the PCs has a practical impact. We give the following example.

Suppose we wish to project another subject onto the plane given by the first two PCs in order to visualize, how the subject's political opinions relate to the deputies and the parties. If the PCs are sparse, the only information that we need is how would the subject vote in the acts that correspond

to nonzero loadings coefficients. Thus we can have the subject respond to only several questions, and the final projection will still be punctual. We reflect on similar considerations further in the Discussion.

Finally, we note that IRLS-S-PCA seems fairly robust regarding the choice of target cardinalities. Varying the target cardinalities around the chosen value alters the resulting figure negligibly. However, if we allow one PC to be described only by less than 10 acts, the plots get noticeably distorted. On the other hand, the number of acts describing one PC can be relatively high but, of course, we aim for the most parsimonious and yet credible representation. As an experimental test of robustness of IRLS-S-PCA, we also tried several times to recalculate the two-dimensional projection with 10 percent of deputies left out and projected onto the subspace subsequently. The results were fairly similar. Also, we would like to note that in a three-dimensional projection, the first two PCs remain very much alike as in the two-dimensional projection. Moreover, the clusters made by parties are still well-separated. We do not report these results due to space restrictions.

DISCUSSION AND CONCLUSION

Our algorithm operates with two tuning parameters: λ and δ. At the moment, we update δ based on a rule of thumb (see default setting in Algorithm 1) and enforce the desired target cardinality heuristically. It would be desirable to understand better the relationship between the tuning parameter λ, regularizing parameter δ, and the resulting cardinality. Such a knowledge would possibly lead to a better strategy for choosing the parameters. Another approach to the choice of λ is to find solutions on a grid of values of λ and pick a value of λ leading to the desired cardinality among them. We believe that methods described in Mazumder et al. (2010) can be employed for this purpose. However, we obtain promising results with the rule of thumb and the simple heuristic, which is computationally less demanding.

In analyses similar to our example with voting records, the number of PCs is often chosen based on the famous "elbow plot" (Jolliffe, 2002), and the values of target cardinality for every PC are chosen such that every sparse PC explains some relatively high proportion of variance explained by the respective standard PC (e.g. Zhang, 2011). We do not recommend this procedure because it leads to too many subsequent decisions based solely on the variance in data. Moreover, this approach simply does not work well for complex data such as the voting records. In the case of our voting records, the previous strategy would lead us to choose 2-3 PCs and the values of target cardinality in hundreds for every PC. Thus we would have much more coefficients to estimate in our model than we can hope to estimate reliably from just 200 observations (deputies). This is the reason why we choose the number of PCs and the respective target cardinalities based on additional considerations, not in a data-driven way.

In the analysis of voting records, we show that SPCA produces better interpretable results than classical PCA. The variable selection property can be utilized to choose important acts. This could be potentially used, for example, to automatically create voting calculators – a popular and at the present manually designed tool that help voters to determine their conformity with different parties and representatives. However, we gave examples why appreciable expert knowledge about actions in the parliament is vital in evaluating the usage of SPCA in this way.

For the voting data, one might naturally wonder whether the sum-of-squares criterion we work with throughout the paper is appropriate. The penalization techniques discussed in Section 1.3 could be readily extended to the framework of penalized likelihood. As a probabilistic model for the votes, one could use for example multinomial distribution. Nonetheless, we restrained from probabilistic model formulation and used sum-of-squares criterion. Thus we view the voting records for deputies as points in standard euclidean space. To our perception, this is somehow natural and,

again, the bottom line is that we delivered promising results with the simplest-to-study sum-of-squares criterion.

ACKNOWLEDGEMENTS

This work was supported by the grants GAČR 15–09663S and SVV 2017 No. 260454.

References

CHEN, X. Adaptive elastic-net sparse principal component analysis for pathway association testing. *Statistical applications in genetics and molecular biology*, 2011, 10(1), pp. 118-134.

CLINTON, J., JACKMAN, S., RIVERS, D. The statistical analysis of roll call data. *American Political Science Review*, 2004, 98(02), pp. 355–370.

DAUBECHIES, I., DEVORE, R., FORNASIER, M., GÜNTÜRK, C. S. Iteratively reweighted least squares minimization for sparse recovery. *Communications on Pure and Applied Mathematics*, 2010, 63(1), pp. 1–38.

DONOHO, D. L. Compressed sensing. *IEEE Transactions on information theory*, 2006, 52(4), pp. 1289–1306.

ECKAR T, C. AND YOUNG, G. The approximation of one matrix by another of lower rank. *Psychometrika*, 1936, 1(3), pp. 211–218.

FAN, J. AND LI, R. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association*, 2001, 96(456), pp. 1348–1360.

HOER L, A. E. AND KENNARD, R. W. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 1970, 12(1), pp. 55–67.

HUNTER, D. R. AND LI, R. Variable selection using mm algorithms. *Annals of statistics*, 2005, 33(4), pp. 1617-1642.

JOHNSTONE, I. M. On the distribution of the largest eigenvalue in principal components analysis. *Annals of statistics*, 2001, pp. 295–327

JOHNSTONE, I. M. AND LU, A. Y. On consistency and sparsity for principal components analysis in high dimensions. *Journal of the American Statistical Association*, 2009, 104(486), pp. 682–693.

JOLLIFFE, I. *Principal component analysis*. John Wiley & Sons, Ltd., 2002.

LANGE, K., HUNTER, D. R., YANG, I. Optimization transfer using surrogate objective functions. *Journal of computational and graphical statistics*, 2000, 9(1), pp. 1–20.

LENG, C. AND WANG, H. On general adaptive sparse principal component analysis. *Journal of Computational and Graphical Statistics*, 2012.

MAZUMDER, R., HASTIE, T., TIBSHIRANI, R. Spectral regularization algorithms for learning large incomplete matrices. *Journal of machine learning research*, 2010, 11(Aug), pp. 2287–2322.

NADLER, B. Finite sample approximation results for principal component analysis: A matrix perturbation approach. *The Annals of Statistics*, 2008, pp. 2791–2817.

PAUL, D. Asymptotics of sample eigenstructure for a large dimensional spiked covariance model. *Statistica Sinica*, 2007, pp. 1617–1642.

SHEN, H. AND HUANG, J. Z. Sparse principal component analysis via regularized low rank matrix approximation. *Journal of multivariate analysis*, 2008, 99(6), pp. 1015–1034.

TIBSHIRANI, R. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 1996, pp. 267–288.

WITTEN, D. M., TIBSHIRANI, R., HASTIE, T. A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistics*, 2009, 10(3), pp. 515–534.

ZHANG, Y. *Sparse principal component analysis: Algorithms and applications*. [PhD. dissertation] UC Berkeley, 2011.

ZOU, H. The adaptive lasso and its oracle properties. *Journal of the American statistical association*, 2006, 101(476), pp. 1418–1429.

ZOU, H. AND HASTIE, T. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2005, 67(2), pp. 301–320.

ZOU, H., HASTIE, T., TIBSHIRANI, R. Sparse principal component analysis. *Journal of computational and graphical statistics*, 2006, 15(2), pp. 265–286.

ZOU, H. AND LI, R. One-step sparse estimates in nonconcave penalized likelihood models. *Annals of statistics*, 2008, 36(4), pp. 1509.

ZOU, H. AND ZHANG, H. H. On the adaptive elastic-net with a diverging number of parameters. *Annals of statistics*, 2009, 37(4), pp. 1733.

APPENDIX

Here we prove Proposition 3. We will use the following two lemmas.

Lemma 4. The following properties hold:

(a) Let $\delta > 0$, $\boldsymbol{\beta} \in \mathbb{R}^p$ and let $w_j = \frac{1}{\beta_j^2+\delta}$, $j = 1, \ldots, p$. Then we have

$$\nabla_{\boldsymbol{\beta}} f(\boldsymbol{\beta}|\delta) = \nabla_{\boldsymbol{\beta}} g(\boldsymbol{\beta}, \boldsymbol{w}|\delta). \tag{20}$$

(b) Let $\delta > 0$ and $x, y \in \mathbb{R}$. Then we have

$$\log(x^2+\delta) - \log(y^2+\delta) - 2\frac{y(x-y)}{x^2+\delta} \geq \frac{(x-y)^2}{x^2+\delta}, \tag{21}$$

and equality in (21) holds if and only if $x = y$.

Proof. For (a), simply calculate the gradients

$$\nabla_{\boldsymbol{\beta}} f(\boldsymbol{\beta}|\delta) = -2\mathbf{X}^\top\big(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\big) + 2\lambda\Big[\frac{\beta_j}{\beta_j^2+\delta}\Big]_{j=1}^p \tag{22}$$

$$\nabla_{\boldsymbol{\beta}} g(\boldsymbol{\beta}, \boldsymbol{w}|\delta) = -2\mathbf{X}^\top\big(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\big) + 2\lambda\big[w_j\beta_j\big]_{j=1}^p \tag{23}$$

and use that $w_j = \frac{1}{\beta_j^2+\delta}$. This gives us (a).

For (b), since $2\frac{y(x-y)}{x^2+\delta} + \frac{(x-y)^2}{x^2+\delta} = \frac{x^2-y^2}{x^2+\delta}$, it is enough to show that

$$\log(x^2+\delta) - \log(y^2+\delta) - \frac{x^2-y^2}{x^2+\delta} \geq 0.$$

For a fixed $x > 0$ is is easy to verify that $h(y) = \log(x^2+\delta) - \log(y^2+\delta) - \frac{x^2-y^2}{x^2+\delta}$ has a unique minimum in $y = x$ by differentiation. The same holds for a fixed $x < 0$ and $x = 0$.

Lemma 5. Under the assumptions of Proposition 3 it holds for $l \in \mathbb{N}$:

$$\big(\mathbf{X}\boldsymbol{\beta}^{(l+1)} - \mathbf{Y}\big)^\top\big(\mathbf{X}\boldsymbol{\beta}^{(l)} - \mathbf{X}\boldsymbol{\beta}^{(l+1)}\big) = -\lambda\sum_{j=1}^p \frac{\beta_j^{(l+1)}(\beta_j^{(l)} - \beta_j^{(l+1)})}{(\beta_j^{(l)})^2 + \delta_l}. \tag{24}$$

Proof. Since $\nabla_{\boldsymbol{\beta}} g(\boldsymbol{\beta}^{(l+1)}, \boldsymbol{w}^{(l)}|\delta) = 0$ from the first order optimality condition, we have

$$\begin{aligned} 0 &= (\boldsymbol{\beta}^{(l)} - \boldsymbol{\beta}^{(l+1)})^\top\nabla_{\boldsymbol{\beta}} g(\boldsymbol{\beta}^{(l+1)}, \boldsymbol{w}^{(l)}|\delta) \\ &= 2\big(\mathbf{X}\boldsymbol{\beta}^{(l+1)} - \mathbf{Y}\big)^\top\big(\mathbf{X}\boldsymbol{\beta}^{(l)} - \mathbf{X}\boldsymbol{\beta}^{(l+1)}\big) + 2\lambda\sum_{j=1}^p \frac{\beta_j^{(l+1)}(\beta_j^{(l)} - \beta_j^{(l+1)})}{(\beta_j^{(l)})^2 + \delta}. \end{aligned}$$

Now we are ready to prove Proposition 3. We calculate for $l \in \mathbb{N}$:

$$\begin{aligned} &f(\boldsymbol{\beta}^{(l)}|\delta_l) - f(\boldsymbol{\beta}^{(l+1)}|\delta_{l+1}) = \\ &= \|\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}^{(l)}\|_2^2 - \|\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}^{(l+1)}\|_2^2 + \lambda\sum_{j=1}^p\big[\log\big((\beta_j^{(l)})^2 + \delta_l\big) - \log\big((\beta_j^{(l+1)})^2 + \delta_{l+1}\big)\big] \\ &= \|\mathbf{X}\boldsymbol{\beta}^{(l)} - \mathbf{X}\boldsymbol{\beta}^{(l+1)}\|_2^2 + 2\big(\mathbf{X}\boldsymbol{\beta}^{(l+1)} - \mathbf{Y}\big)^\top\big(\mathbf{X}\boldsymbol{\beta}^{(l)} - \mathbf{X}\boldsymbol{\beta}^{(l+1)}\big) \\ &\qquad + \lambda\sum_{j=1}^p\big[\log\big((\beta_j^{(l)})^2 + \delta_l\big) - \log\big((\beta_j^{(l+1)})^2 + \delta_{l+1}\big)\big] \\ &\geq \lambda\sum_{j=1}^p\big[\log\big((\beta_j^{(l)})^2 + \delta_l\big) - \log\big((\beta_j^{(l+1)})^2 + \delta_{l+1}\big) - 2\tfrac{\beta_j^{(l+1)}(\beta_j^{(l)} - \beta_j^{(l+1)})}{(\beta_j^{(l)})^2+\delta_l}\big] \\ &\geq \lambda\sum_{j=1}^p \tfrac{(\boldsymbol{\beta}^{(l+1)} - \boldsymbol{\beta}^{(l)})^2}{(\boldsymbol{\beta}^{(l)})^2+\delta_l} \geq \lambda\sum_{j=1}^p \tfrac{(\boldsymbol{\beta}^{(l+1)} - \boldsymbol{\beta}^{(l)})^2}{(\boldsymbol{\beta}^{(l)})^2+\delta_0}. \end{aligned}$$

The second equality is just a tedious algebra. In the first inequality, the squared ℓ_2-norm is dropped and Lemma 5 is used. In the second inequality, δ_{l+1} is substituted for greater δ_l reducing the overall value of the expression, and then Lemma 4 (b) is used. In the last inequality, δ_l is substituted again for a greater δ_0.

Note the previous calculations suggest that $f(\boldsymbol{\beta}^{(l)}|\delta_l) \geq f(\boldsymbol{\beta}^{(l+1)}|\delta_{l+1})$ for all $l \in \mathbb{N}$. Thus for all $l \in \mathbb{N}$ we have

$$f(\boldsymbol{\beta}^{(1)}|\delta_1) \geq f(\boldsymbol{\beta}^{(l)}|\delta_l) \geq \lambda \sum_{j=1}^{p} \log\left((\beta_j^{(l)})^2 + \delta_l\right) \geq \lambda \sum_{j=1}^{p} \log\left((\beta_j^{(l)})^2 + \delta_\star\right), \tag{25}$$

from which it follows that there exists a constant c such that for all $l \in \mathbb{N}$ and $j = 1, \ldots, p$ it holds $|\beta_j^{(l)}| \leq c$. It follows that

$$f(\boldsymbol{\beta}^{(l)}|\delta_l) - f(\boldsymbol{\beta}^{(l+1)}|\delta_{l+1}) \geq \frac{\lambda}{c^2 + \delta_0} \sum_{j=1}^{p} (\boldsymbol{\beta}^{(l+1)} - \boldsymbol{\beta}^{(l)})^2 = \frac{\lambda}{c^2 + \delta_0} \|\boldsymbol{\beta}^{(l+1)} - \boldsymbol{\beta}^{(l)}\|_2^2. \tag{26}$$

It also follows from (25) that $\{f(\boldsymbol{\beta}^{(l)}|\delta_l)\}_{l=1}^{\infty}$ converges for $l \to \infty$ monotically to some finite value, let us label it $f^\star$. Thus summing (26) for $l = 1, 2, \ldots$ we obtain

$$f(\boldsymbol{\beta}^{(1)}|\delta_1) - f^\star \geq \frac{\lambda}{c + \delta_0} \sum_{l=1}^{\infty} \|\boldsymbol{\beta}^{(l+1)} - \boldsymbol{\beta}^{(l)}\|_2^2,$$

from which it follows

$$\|\boldsymbol{\beta}^{(l+1)} - \boldsymbol{\beta}^{(l)}\|_2 \to 0. \tag{27}$$

Next, let us have $\{\boldsymbol{\beta}^{(l_m)}\}_{m=1}^{\infty} \subset \{\boldsymbol{\beta}^{(l)}\}_{l=1}^{\infty}$ such that $\boldsymbol{\beta}^{(l_m)} \to \boldsymbol{\beta}^\star$. Existence of this convergent subsequence follows from compactness of the level sets of functions $f(\boldsymbol{\beta}|\delta)$, $\delta > 0$. Then

$$w_j^{(l_m)} = \frac{1}{(\beta_j^{(l_m)})^2 + \delta_{l_m}} \to \frac{1}{(\beta_j^\star)^2 + \delta_\star} = w_j^\star$$

and from (27) also $\boldsymbol{\beta}^{(l_m+1)} \to \boldsymbol{\beta}^\star$. By taking the limit of the first order optimality condition $\nabla_{\boldsymbol{\beta}}\, g(\boldsymbol{\beta}^{(l_m+1)}, \boldsymbol{w}^{(l_m)}|\delta_{l_m}) = 0$ we obtain $\nabla_{\boldsymbol{\beta}}\, g(\boldsymbol{\beta}^\star, \boldsymbol{w}^\star|\delta_\star) = 0$, and by using Lemma 4 (a) we have $\nabla f(\boldsymbol{\beta}^\star|\delta_\star) = 0$, which completes the proof.