Optimization-based data analysis Fall 2017

Lecture Notes 8: Convex Nondifferentiable Functions

1 Applications

1.1 Sparse regression

In our description of linear regression in Lecture Notes 6, we assume implicitly that all features are related to the response. However, this is often not the case in applications: some measured features may be unrelated and should not be included in the model. Selecting relevant features is a crucial problem in statistics, which is known as *model selection*. In this section, we consider the problem of selecting a small subset of relevant features that yield a good linear approximation to the data. This is equivalent to finding a *sparse* vector of coefficients $\vec{\beta}$ such that

$$y^{(i)} \approx \left\langle \vec{x}^{\,(i)}, \vec{\beta} \right\rangle. \tag{1}$$

The number of selected features is equal to the number of nonzero entries in $\vec{\beta}$.

When fitting a sparse linear model we have two objectives:

- Achieving a good fit to the data; $\left|\left|X\vec{\beta} - \vec{y}\right|\right|_2^2$ should be as small as possible.
- Using a small number of features; $\vec{\beta}$ should be as sparse as possible.

This suggests minimizing a cost function that simultaneously promotes a good fit to the data and sparsity in the vector of coefficients. In Lecture Notes 6 we describe the ridge-regression estimator, that uses an ℓ_2-norm regularization term to ensure that the norm of the coefficients is not too large. Here we would like to control the number of nonzeros in the support of the coefficient, i.e. its ℓ_0 "norm" instead. However, this "norm" is not convex and very difficult to minimize (see Example 1.8 in Lecture Notes 7). Instead, we incorporate an ℓ_1-norm regularization that promotes sparsity, but is still convex. In statistics, the solution to an ℓ_1-norm-regularized least-squares problem is called the *lasso* estimator, introduced in [9] (see also [6]).

Definition 1.1 (The lasso). *For $X \in \mathbb{R}^{n\times p}$ and $\vec{y} \in \mathbb{R}^p$, the lasso estimate is the minimizer of the optimization problem*

$$\vec{\beta}_{\text{lasso}} := \arg\min_{\vec{\beta}} \frac{1}{2}\left|\left|\vec{y} - X\vec{\beta}\right|\right|_2^2 + \lambda\left|\left|\vec{\beta}\right|\right|_1, \tag{2}$$

where $\lambda > 0$ is a fixed regularization parameter.

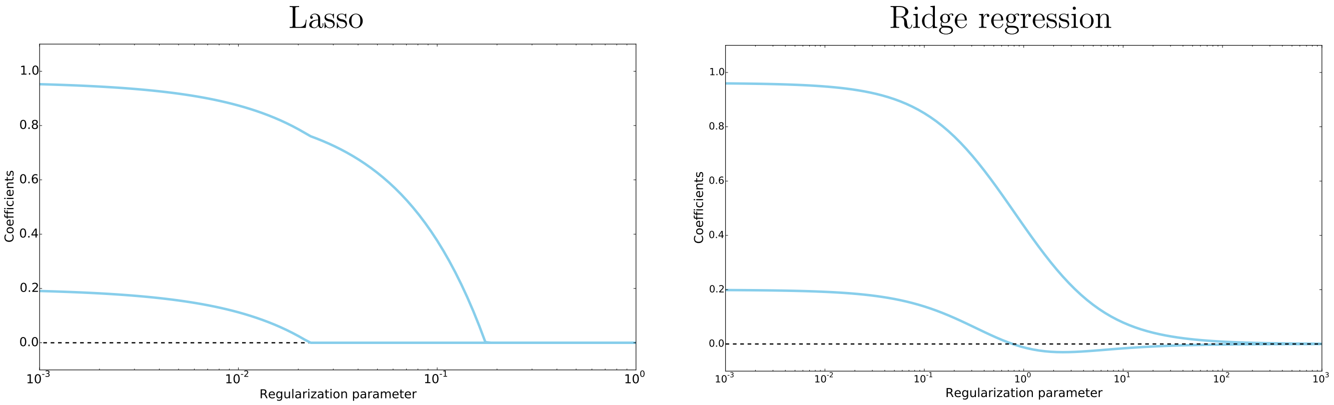

Figure 1: Coefficients of the lasso and ridge-regression estimates in the sparse regression problem in Example 1.4 for $\alpha = 1$, 5 examples ($n = 5$), $\rho := -0.43$ and different values of the regularization parameter λ.

The following lemma shows that sums of convex functions are convex, so the lasso cost function is indeed convex.

Lemma 1.2 (Nonnegative weighted sums)**.** *The weighted sum of m convex functions $f_1, \ldots, f_m$*

$$f := \sum_{i=1}^{m} \alpha_i \, f_i \tag{3}$$

is convex as long as the weights $\alpha_1, \ldots, \alpha \in \mathbb{R}$ are nonnegative.

Proof. By convexity of $f_1, \ldots, f_m$, for any $\vec{x}, \vec{y} \in \mathbb{R}^m$ and any $\theta \in (0, 1)$

$$\begin{aligned} f\left(\theta \vec{x} + (1-\theta)\,\vec{y}\right) &= \sum_{i=1}^{m} \alpha_i \, f_i\left(\theta \vec{x} + (1-\theta)\,\vec{y}\right) && (4) \\ &\leq \sum_{i=1}^{m} \alpha_i \left(\theta f_i\left(\vec{x}\right) + (1-\theta)\, f_i\left(\vec{y}\right)\right) && (5) \\ &= \theta\, f\left(\vec{x}\right) + (1-\theta)\, f\left(\vec{y}\right). && (6) \end{aligned}$$

□

Corollary 1.3 (Regularized least squares)**.** *Regularized least-squares cost functions of the form*

$$\left|\left|A\vec{x} - \vec{y}\right|\right|_2^2 + \left|\left|\vec{x}\right|\right|, \tag{7}$$

where $\left|\left|\cdot\right|\right|$ is an arbitrary norm, are convex.

Example 1.4 (Sparse regression with two features)**.** We consider a simple sparse regression problem where the response only depends on one feature,

$$\vec{y} := \alpha \vec{x}_1 + \vec{z}, \tag{8}$$

where $\vec{y} \in \mathbb{R}^n$ is the response vector, $\vec{x} \in \mathbb{R}^n$ contains the relevant feature and $\vec{z} \in \mathbb{R}^n$ is additive noise. In our data set, there are two features $\vec{x}_1$ and $\vec{x}_2$, which is irrelevant to the response. However, we don't know this a priori, so we use the feature matrix

$$X := \begin{bmatrix} \vec{x}_1 & \vec{x}_2 \end{bmatrix} \tag{9}$$

to fit a linear-regression model with both features by minimizing the least-squares cost function. Both features are normalized so that $||\vec{x}_1||_2 = ||\vec{x}_2||_2 = 1$. The correlation between them equals

$$\rho := \langle \vec{x}_1, \vec{x}_2 \rangle \,. \tag{10}$$

Unfortunately, the least-square estimate of the vector of coefficients is dense

$$\begin{aligned}
\vec{\beta}_{\text{LS}} &= \left(X^T X\right)^{-1} X^T \vec{y} && (11)\\
&= \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}^{-1} \begin{bmatrix} \vec{x}_1^T \vec{y} \\ \vec{x}_2^T \vec{y} \end{bmatrix} && (12)\\
&= \frac{1}{1-\rho^2} \begin{bmatrix} 1 & -\rho \\ -\rho & 1 \end{bmatrix} \begin{bmatrix} \alpha + \vec{x}_1^T \vec{z} \\ \alpha\rho + \vec{x}_2^T \vec{z} \end{bmatrix} && (13)\\
&= \begin{bmatrix} \alpha \\ 0 \end{bmatrix} + \frac{1}{1-\rho^2} \begin{bmatrix} \langle \vec{x}_1 - \rho \vec{x}_2, \vec{z} \rangle \\ \langle \vec{x}_2 - \rho \vec{x}_1, \vec{z} \rangle \end{bmatrix} && (14)
\end{aligned}$$

unless the noise happens to be orthogonal to both $\vec{x}_1$ and $\vec{x}_2$, which is not the case with high probability. Ridge regression also produces a dense estimate. In contrast, the lasso estimate is sparse and correctly identifies the right feature. The value of the coefficients for the ridge-regression and lasso estimators are shown in Figure 1 for $\alpha = 1$, 5 examples ($n = 5$) and $\rho := -0.43$. For large λ both estimators are equal to zero, as the regularization term dominates. For small λ the estimators tend to the least-squares estimators. For intermediate values of λ, the ℓ_1-norm regularization term promotes a sparse coefficient vector, whereas the ℓ_2-norm regularization term does not. △

Example 1.5 (Prostate cancer data set)**.** In this example, we apply the lasso to a sparse regression problem related to the study of prostate cancer.[1] The response is the level of prostate-specific antigen (PSA) measured for a specific patient (high levels of PSA are an indication of cancer), whereas the features are characteristics of the patient, including age, weight and other measurements. We fit a sparse linear model to the data using the lasso. The training set contains 60 patients, whereas the test set contains 37 patients. Figure 2 shows the coefficients for different values of the regularization parameter λ. The least-squares estimate ($\lambda \to 0$) achieves the smallest ℓ_2 error on the training set using all of the features. The lasso estimate with λ between 0.1 and 0.5 incurs in a larger training error but achieves a similar, or better test error, with a smaller number of coefficients (5 instead of 8), suggesting that the 3 remaining features are not related to the response. △

[1] The data is available here.

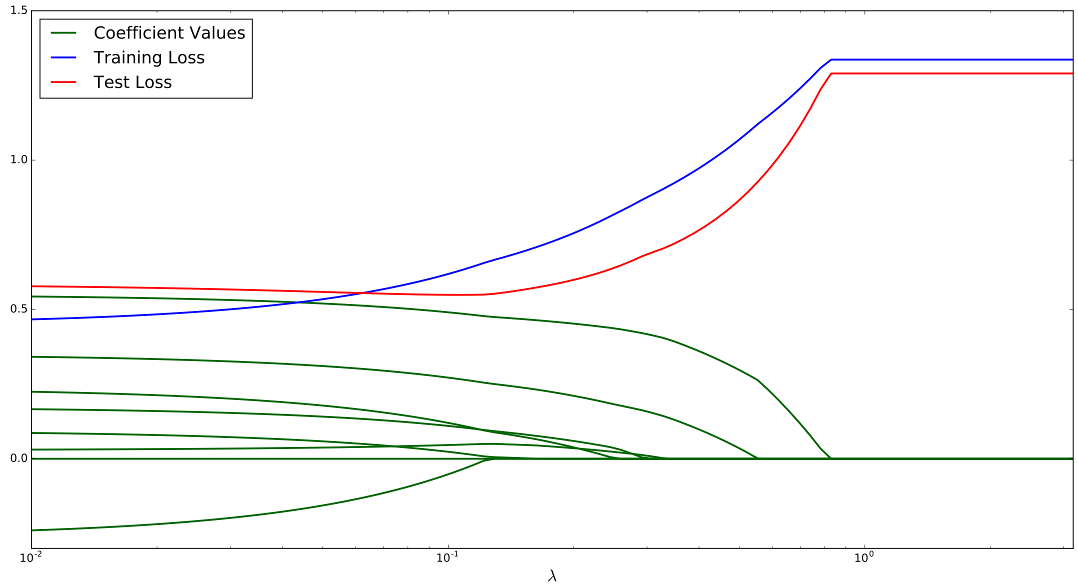

Figure 2: Coefficients, training error and test error of the lasso estimate for different values of the regularization parameter λ when applied to the sparse-regression problem in Example 1.5.

1.2 Robust principal-component analysis

Outliers may severely distort the results of applying principal-component analysis (PCA) to a set of data that lie close to a low-dimensional subspace. Even one outlier can have a significant effect, as illustrated in the following example.

Example 1.6 (PCA with an outlier)**.** A data set contains five examples with three features each. We apply PCA to these data by computing the SVD of the matrix

$$Y := \begin{bmatrix} -2 & -1 & 5 & 1 & 2 \\ -2 & -1 & 0 & 1 & 2 \\ -2 & -1 & 0 & 1 & 2 \end{bmatrix}. \tag{15}$$

All data points are aligned with the vector $\begin{bmatrix} 1 & 1 & 1 \end{bmatrix}^T$ except for the third one, due to an outlier that has corrupted one of the features (shown in red). Due of this outlier, the data matrix has rank 2 instead of 1 and the principal directions are not aligned on the line that contains most of the points. Figure 3 shows the data points, as well as the principal directions when the outlier is present and when it is absent. $\triangle$

This is an example of a general phenomenon where a small number of corrupted entries disrupts low-rank structure in a matrix, making it appear high rank, despite the correlations between columns (or rows). As a result, computing the SVD does not uncover the low-rank component of the data. An alternative is to fit a low rank + sparse model to the data, where the sparse component accounts for the outliers and the low-rank component reveals the underlying correlations.

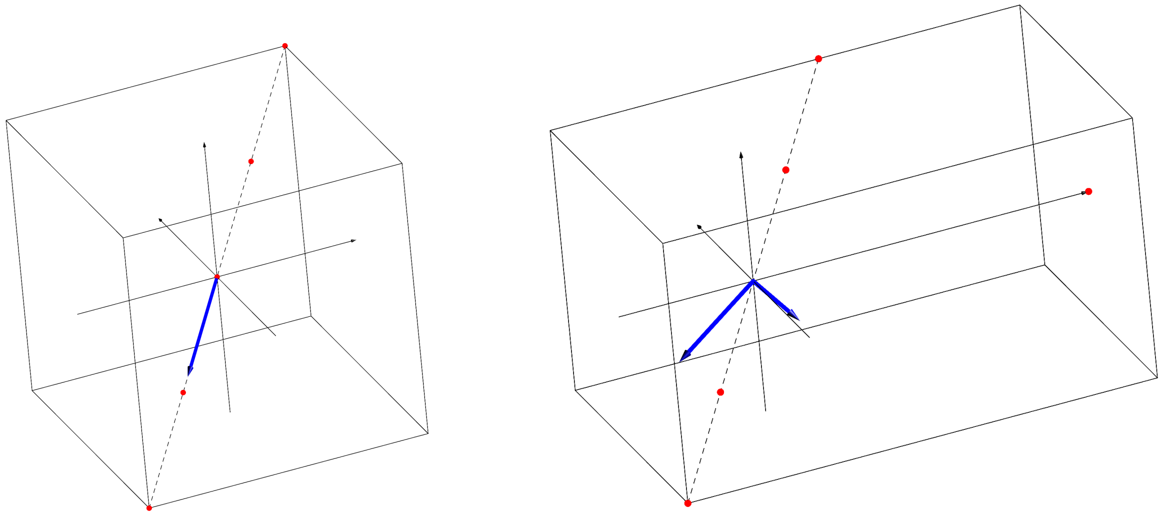

Figure 3: The data points in Example 1.6 are plotted in red. On the left, the data contains no outliers and the principal direction (blue) corresponding to the only nonzero singular value of the SVD of the data matrix is aligned with all the points. On the right, adding the outlier distorts the principal directions (in blue), which are two instead of one because the rank of the matrix increases by one.

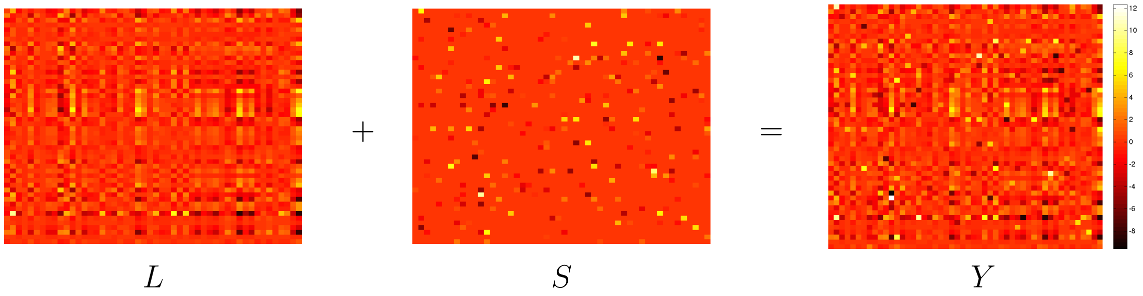

Figure 4: Y is obtained by summing a low-rank matrix L and a sparse matrix S.

Figure 4 shows a simulated example of this model. As illustrated in Examples 1.8 and 1.10, it is usually not tractable to maximize sparsity and minimize rank directly. An alternative that often works well is to penalize the ℓ_1 and nuclear norm respectively. This technique introduced by [3,5] is often called *robust PCA* (RPCA), since it aims to obtain a low-rank component that is not affected by the presence of outliers.

Algorithm 1.7 (Robust PCA)**.** *For $Y \in \mathbb{R}^{n \times m}$, the robust PCA estimator of the low-rank component in Y is the minimizer of the optimization problem*

$$L_{\text{RPCA}} := \arg\min_{L} ||L||_* + \lambda \, ||Y - L||_1 \,, \tag{16}$$

where $\lambda > 0$ is a fixed regularization parameter. Here $||\cdot||_1$ denotes the sum of absolute values of the entries of the matrix; it is the ℓ_1 norm of the vectorized matrix. $S_{\text{RPCA}} := Y - L_{\text{RPCA}}$ is the estimator of the sparse component.

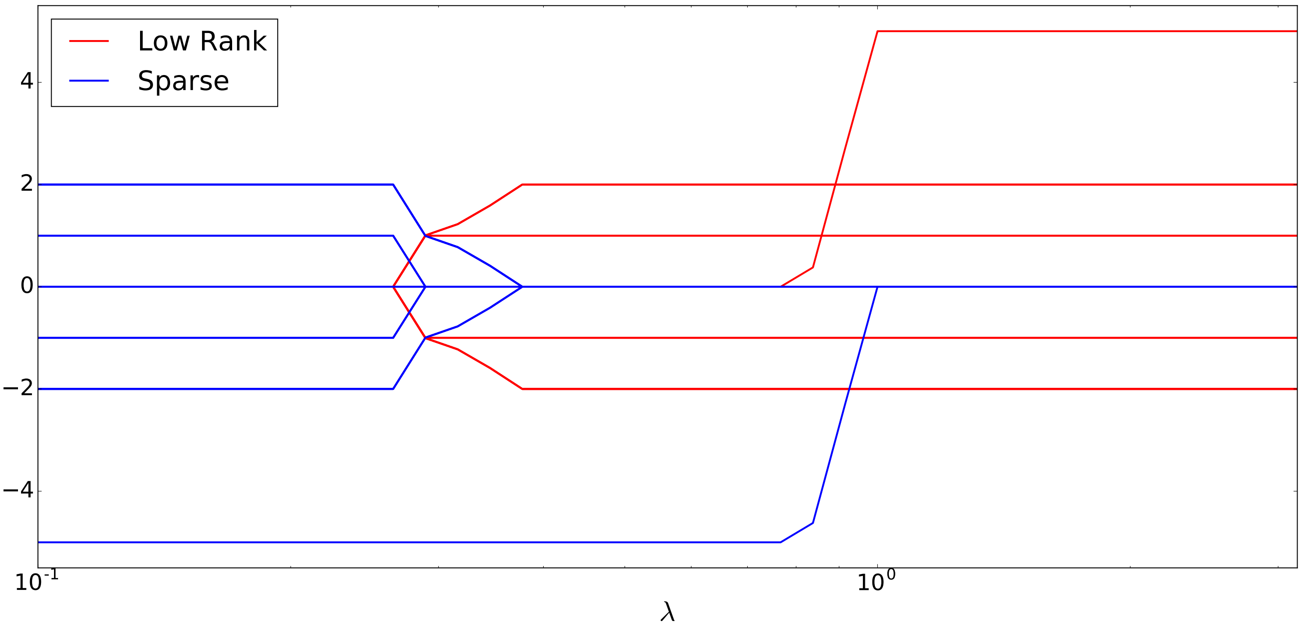

Figure 5: Values of the entries of the low-rank and sparse components for different values of λ computed by applying RPCA to the data in Example 1.6.

Once the low-rank component has been recovered, PCA can be applied to it to determine its principal directions. Figure 5 shows the result of applying RPCA to the data in Example 1.6. If the parameter λ is in certain range, then the low-rank component exactly uncovers the rank-1 structure in the data and the sparse component identifies the outlier. In general, the regularization parameter λ determines the weight of the two structure-inducing terms in the cost function. Figure 6 shows the low-rank and sparse components of the matrix in Figure 4 for different values of λ. If λ is too small, then it is *cheap* to increase the content of the sparse component, which consequently won't be very sparse. Similarly, if λ is too large, then the low-rank component won't be low-rank, as the nuclear-norm term has less influence. Setting λ correctly allows to achieve a perfect decomposition. In practice, the regularization parameter is usually set using cross validation.

Example 1.8 (Background subtraction)**.** In computer vision, the problem of background subtraction is that of separating the background and foreground of a video sequence. In particular we consider a scene with a static background. If we stack the video frames in a matrix Y, where each column corresponds to a vectorized frame, and the background is completely static, then all the frames are equal to a certain vector $\vec{x} \in \mathbb{R}^m$ (m is the number of pixels in each frame) and the matrix is rank 1,

$$Y = \begin{bmatrix} \vec{x} & \vec{x} & \cdots & \vec{x} \end{bmatrix} = \vec{x} \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix}. \tag{17}$$

If the background is not completely static, but instead experiences gradual changes, then the matrix containing the frames will be approximately low rank. If there are sudden events in the foreground that occupy a small part of the field of view and do not last very long, then this is equivalent to adding a sparse component (most entries are equal to zero) to the low-rank background. The two components can consequently be separated using the robust PCA algorithm. The results of applying this method to a real video sequence are shown in Figure 7. △

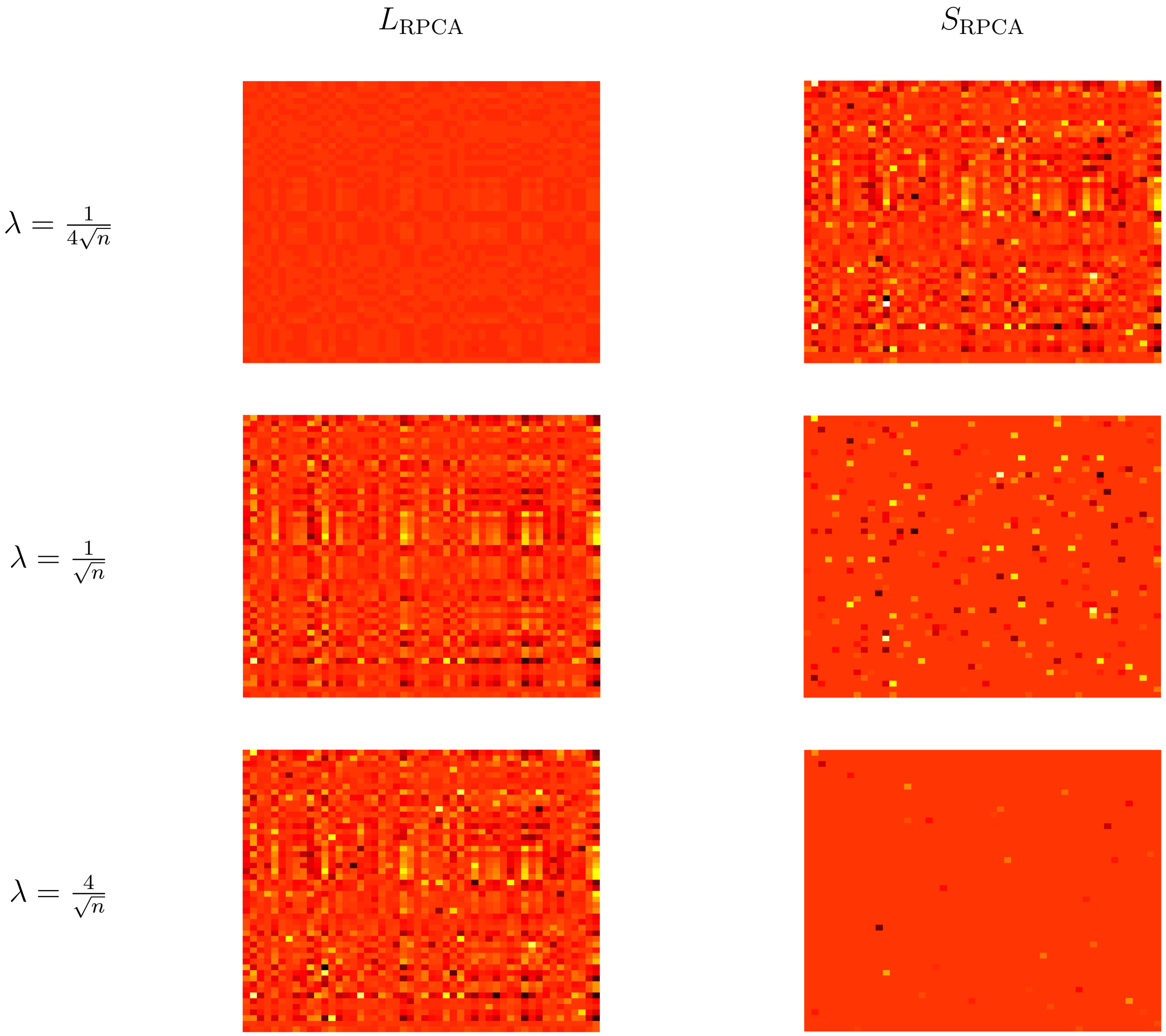

Figure 6: RPCA estimates of the low-rank and sparse components of the matrix in Figure 4 for different values of the regularization parameter. For $\lambda := 1/\sqrt{n}$ the components are recovered perfectly.

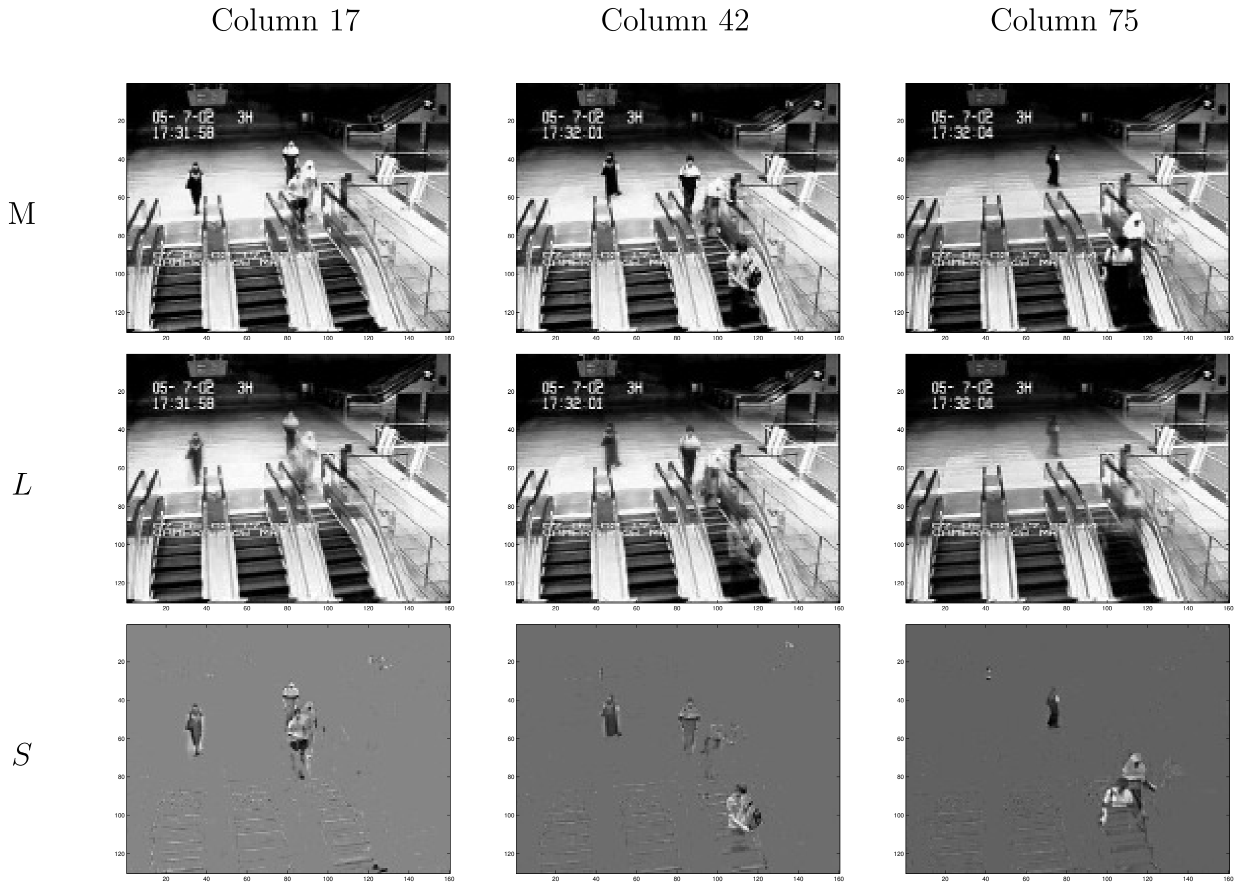

Figure 7: Background subtraction results from a video. This example is due to Stephen Becker. The code is available at http://cvxr.com/tfocs/demos/rpca.

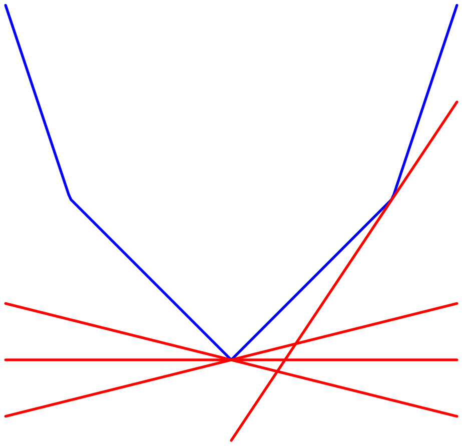

Figure 8: A nondifferentiable convex function (blue). The red supporting lines are specified by subgradients that determine their slope.

2 Subgradients

2.1 Definition and properties

By Theorem 2.5 in Lecture Notes 7, differentiable functions are convex if and only if their epigraph has a supporting hyperplane at every point. In more detail, a differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if and only at any point $\vec{x} \in \mathbb{R}^n$ there exists a vector $\vec{g}$ such that

$$f(\vec{y}) \geq f(\vec{x}) + \langle \vec{g}, \vec{y} - \vec{x} \rangle \tag{18}$$

for every $\vec{y} \in \mathbb{R}^n$. In the case of differentiable functions, $\vec{g}$ is the gradient of f at $\vec{x}$. Nondifferentiable functions do not have gradients, but the existence of a supporting hyperplane still characterizes convexity. The vector $\vec{g}$ that corresponds to such a hyperplane is called a subgradient.

Definition 2.1 (Subgradient)**.** *The subgradient of a function $f : \mathbb{R}^n \to \mathbb{R}$ at $\vec{x} \in \mathbb{R}^n$ is a vector $\vec{g} \in \mathbb{R}^n$ such that*

$$f(\vec{y}) \geq f(\vec{x}) + \vec{g}^{\,T}(\vec{y} - \vec{x}), \quad \textit{for all } \vec{y} \in \mathbb{R}^n. \tag{19}$$

The set of all subgradients is called the subdifferential of the function at $\vec{x}$.

Figure 8 shows a one-dimensional nondifferentiable convex function, along with some of the hyperplanes that support its epigraph. The following theorem establishes that a function is convex if and only if a subgradient exists at every point.

Theorem 2.2 (Proof in Section 4.1)**.** *A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if and only if it has a non-empty subdifferential at any $\vec{x} \in \mathbb{R}^n$. It is strictly convex if and only for all $\vec{x} \in \mathbb{R}^n$ there exists a subgradient $\vec{g} \in \mathbb{R}^n$ such that*

$$f(\vec{y}) \geq f(\vec{x}) + \vec{g}^{\,T}(\vec{y} - \vec{x}), \quad \textit{for all } \vec{y} \in \mathbb{R}^n. \tag{20}$$

Subgradients are a useful tool to characterize the minima of nondifferentiable convex functions.

Theorem 2.3 (Optimality condition)**.** *A convex function attains its minimum value at a vector x if and only if the zero vector is a subgradient of f at x. If the function is strictly convex, then the minimum is unique.*

Proof. By the definition of subgradient, if $\vec{g} := \vec{0}$ is a subgradient at $\vec{x}$, then for any $\vec{y} \in \mathbb{R}^n$

$$f(\vec{y}) \geq f(\vec{x}) + \vec{g}^T(\vec{y} - \vec{x}) = f(\vec{x}), \tag{21}$$

which is equivalent to $\vec{x}$ being a global minimum of the function. If the function is strictly convex, then the inequality is strict for all $\vec{y} \neq \vec{x}$. □

A useful property is that the sum of subgradients of two or more functions is a subgradient of their sum.

Lemma 2.4 (Sum of subgradients)**.** *Let $\vec{g}_1$ and $\vec{g}_2$ be subgradients at $\vec{x} \in \mathbb{R}^n$ of $f_1 : \mathbb{R}^n \to \mathbb{R}$ and $f_2 : \mathbb{R}^n \to \mathbb{R}$ respectively. Then $\vec{g} := \vec{g}_1 + \vec{g}_2$ is a subgradient of $f := f_1 + f_2$ at $\vec{x}$.*

Proof. For any $\vec{y} \in \mathbb{R}^n$

$$\begin{aligned} f(\vec{y}) &= f_1(\vec{y}) + f_2(\vec{y}) && (22)\\ &\geq f_1(\vec{x}) + \vec{g}_1^T(\vec{y} - \vec{x}) + f_2(\vec{y}) + \vec{g}_2^T(\vec{y} - \vec{x}) && (23)\\ &\geq f(\vec{x}) + \vec{g}^T(\vec{y} - \vec{x}). && (24) \end{aligned}$$

□

Another useful property is that the subgradient of a function scaled by a constant can be obtained by scaling the subgradient.

Lemma 2.5 (Subgradient of scaled function)**.** *Let $\vec{g}_1$ be a subgradient at $\vec{x} \in \mathbb{R}^n$ of $f_1 : \mathbb{R}^n \to \mathbb{R}$. Then for any nonnegative $\eta \in \mathbb{R}$ $\vec{g}_2 := \eta \vec{g}_1$ is a subgradient of $f_2 := \eta f_1$ at $\vec{x}$.*

Proof. For any $\vec{y} \in \mathbb{R}^n$

$$\begin{aligned} f_2(\vec{y}) &= \eta f_1(\vec{y}) && (25)\\ &\geq \eta\left(f_1(\vec{x}) + \vec{g}_1^T(\vec{y} - \vec{x})\right) && (26)\\ &\geq f_2(\vec{x}) + \vec{g}_2^T(\vec{y} - \vec{x}). && (27) \end{aligned}$$

□

2.2 Examples of subdifferentials

If a function is differentiable at a given point, then the gradient is the only subgradient at that point.

Theorem 2.6 (Subdifferential of differentiable functions)**.** *If a convex function $f : \mathbb{R}^n \to \mathbb{R}$ is differentiable at $\vec{x} \in \mathbb{R}^n$, then its subdifferential at $\vec{x}$ only contains $\nabla f(\vec{x})$.*

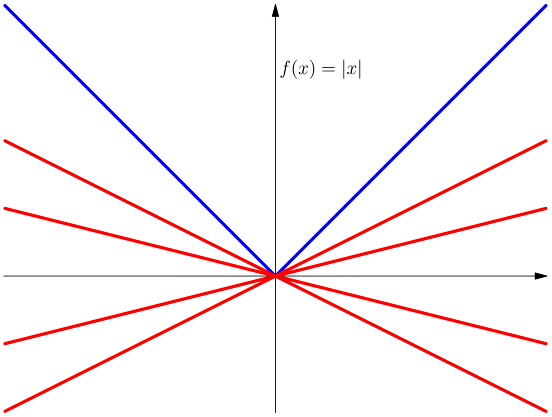

Figure 9: Examples of supporting lines of the absolute value function at the origin. The subgradients at the origin determine the slope of the lines.

Proof. By Theorem 2.5 in Lecture Notes 7 $\nabla f(\vec{x})$ is a subgradient at $\vec{x}$. Now, let $\vec{g}$ be an arbitrary subgradient at $\vec{x}$. By the definition of subgradient, for any $1 \leq i \leq n$

$$\begin{aligned} f(\vec{x} + \alpha \vec{e}_i) &\geq f(\vec{x}) + \vec{g}^T \alpha \vec{e}_i && (28) \\ &= f(\vec{x}) + \vec{g}[i] \alpha, && (29) \\ f(\vec{x}) &\geq f(\vec{x} - \alpha \vec{e}_i) + \vec{g}^T \alpha \vec{e}_i && (30) \\ &= f(\vec{x} - \alpha \vec{e}_i) + \vec{g}[i] \alpha, && (31) \end{aligned}$$

where $\vec{e}_i$ is the ith vector in the standard basis (all its entries are equal to zero, except the ith entry which is equal to one). Combining both inequalities

$$\frac{f(\vec{x}) - f(\vec{x} - \alpha \vec{e}_i)}{\alpha} \leq \vec{g}[i] \leq \frac{f(\vec{x} + \alpha \vec{e}_i) - f(\vec{x})}{\alpha}. \tag{32}$$

If we let $\alpha \to 0$, this implies $\vec{g}[i] = \frac{\partial f(\vec{x})}{\partial \vec{x}[i]}$. Consequently, $\vec{g} = \nabla f(\vec{x})$. □

The following lemma characterizes the subdifferential of the absolute value function.

Lemma 2.7 (Subdifferential of absolute value)**.** *The subdifferential of the absolute value function* $|\cdot| : \mathbb{R} \to \mathbb{R}$ *at* x *is equal to* $\{\text{sign}(x)\}$ *if* $x \neq 0$ *and to* $\{g \in \mathbb{R} \mid |g| \leq 1\}$ *if* $x = 0$.

Proof. If $x \neq 0$ the function is differentiable and the only subgradient is equal to the derivative by Theorem 2.6. At $x = 0$, we need

$$\begin{aligned} |y| &= f(0 + y) && (33) \\ &\geq f(0) + g(y - 0) && (34) \\ &\geq gy && (35) \end{aligned}$$

for all $y \in \mathbb{R}$, which holds if and only if $|g| \leq 1$. □

As motivated in Section 1.1, the ℓ_1 norm is an important nondifferentiable convex function in data analysis. The following theorem characterizes its subdifferential.

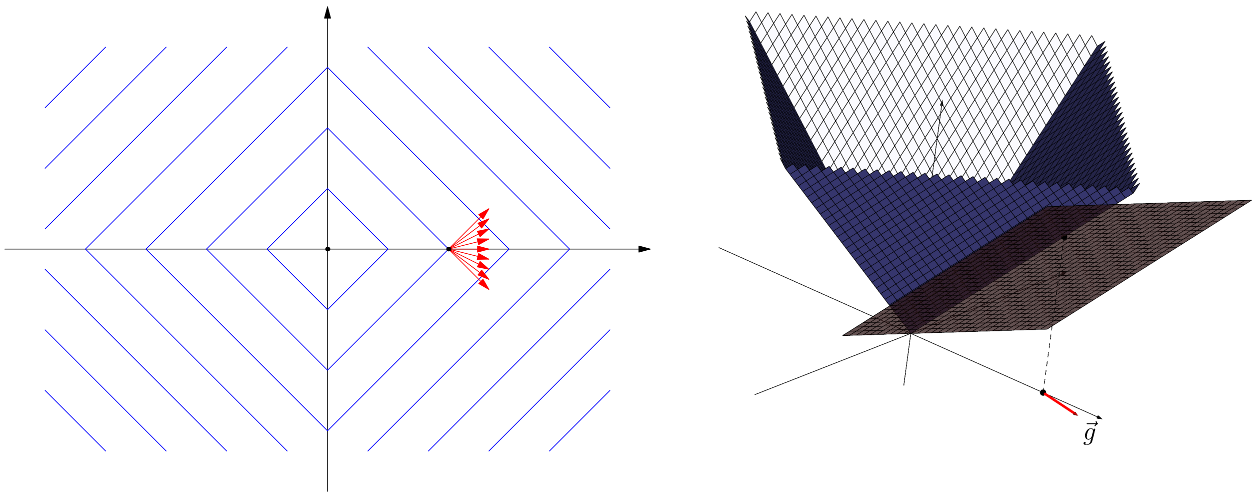

Figure 10: On the left the blue lines are contour lines of the ℓ_1 norm in $\mathbb{R}^2$. The red arrows correspond to subgradients at a point where the function is nondifferentiable. On the right the graph of the function is shown in blue, and the supporting hyperplane corresponding to one of the subgradients (denoted by $\vec{g}$ and a red line) is plotted in brown.

Theorem 2.8 (Subdifferential of ℓ_1 norm)**.** *The subdifferential of the ℓ_1 norm at $\vec{x} \in \mathbb{R}^n$ is the set of vectors $\vec{g} \in \mathbb{R}^n$ that satisfy*

$$\vec{g}[i] = \operatorname{sign}\left(\vec{x}[i]\right) \quad \textit{if } \vec{x}[i] \neq 0, \tag{36}$$

$$|\vec{g}[i]| \leq 1 \quad \textit{if } \vec{x}[i] = 0. \tag{37}$$

The theorem is a direct consequence of Lemma 2.7 and the following result.

Lemma 2.9. *The vector $\vec{g} \in \mathbb{R}^n$ is a subgradient of $||\cdot||_1 : \mathbb{R}^n \to \mathbb{R}$ at $\vec{x}$ if and only if $q[i]$ is a subgradient of $|\cdot| : \mathbb{R} \to \mathbb{R}$ at $\vec{x}[i]$ for all $1 \leq i \leq n$.*

Proof. If $\vec{g}$ is a subgradient of $||\cdot||_1$ at $\vec{x}$ then for any $y \in \mathbb{R}$

$$|y| = |\vec{x}[i]| + ||\vec{x} + (y - \vec{x}[i])\,\vec{e}_i||_1 - ||\vec{x}||_1 \tag{38}$$

$$\geq |\vec{x}[i]| + ||\vec{x}||_1 + \vec{g}^{\,T}(y - \vec{x}[i])\,\vec{e}_i - ||\vec{x}||_1 \tag{39}$$

$$= |\vec{x}[i]| + \vec{g}[i]\,(y - \vec{x}[i])\,, \tag{40}$$

so $\vec{g}[i]$ is a subgradient of $|\cdot|$ at $|\vec{x}[i]|$ for any $1 \leq i \leq n$.

If $\vec{g}[i]$ is a subgradient of $|\cdot|$ at $|\vec{x}[i]|$ for $1 \leq i \leq n$ then for any $\vec{y} \in \mathbb{R}^n$

$$||\vec{y}||_1 = \sum_{i=1}^{n} |\vec{y}\,[i]| \tag{41}$$

$$\geq \sum_{i=1}^{n} |\vec{x}[i]| + \vec{g}[i]\,(\vec{y}\,[i] - \vec{x}[i]) \tag{42}$$

$$= ||\vec{x}||_1 + \vec{g}^{\,T}(\vec{y} - \vec{x}) \tag{43}$$

so $\vec{g}$ is a subgradient of $||\cdot||_1$ at $\vec{x}$. □

Another important nondifferentiable convex function in data analysis is the nuclear norm (see Section 1.2). The following theorem characterizes its subdifferential.

Theorem 2.10 (Subdifferential of the nuclear norm)**.** *Let $X \in \mathbb{R}^{m\times n}$ be a rank-r matrix with SVD USV^T, where $U \in \mathbb{R}^{m\times r}$, $V \in \mathbb{R}^{n\times r}$ and $S \in \mathbb{R}^{r\times r}$ contains the nonzero singular values of X. The subdifferential of the nuclear norm at X is the set of matrices of the form*

$$G := UV^T + W \tag{44}$$

where W satisfies

$$\begin{aligned} \|W\| &\leq 1, && (45)\\ U^T W &= 0, && (46)\\ W\,V &= 0. && (47) \end{aligned}$$

Proof. We only prove that a matrix of the form (44) is a valid subgradient. For the converse (all subgradients are of this form) see [12]. By Pythagoras' Theorem, for any $\vec{x} \in \mathbb{R}^m$ with unit ℓ_2 norm we have

$$\begin{aligned} \left|\left|\mathcal{P}_{\text{row}(X)}\,\vec{x}\right|\right|_2^2 + \left|\left|\mathcal{P}_{\text{row}(X)^\perp}\,\vec{x}\right|\right|_2^2 &= \|\vec{x}\|_2^2 && (48)\\ &= 1. && (49) \end{aligned}$$

As result, since the rows of UV^T are all in $\text{row}\,(X)$ and the rows of W are in $\text{row}\,(X)^\perp$ by Condition (47)

$$\begin{aligned} \|G\|^2 &:= \max_{\{\|\vec{x}\|_2=1\ |\ \vec{x}\in\mathbb{R}^n\}} \|G\,\vec{x}\|_2^2 && (50)\\ &= \max_{\{\|\vec{x}\|_2=1\ |\ \vec{x}\in\mathbb{R}^n\}} \left|\left|UV^T\,\vec{x}\right|\right|_2^2 + \|W\,\vec{x}\|_2^2 && (51)\\ &= \max_{\{\|\vec{x}\|_2=1\ |\ \vec{x}\in\mathbb{R}^n\}} \left|\left|UV^T\,\mathcal{P}_{\text{row}(X)}\,\vec{x}\right|\right|_2^2 + \left|\left|W\,\mathcal{P}_{\text{row}(X)^\perp}\,\vec{x}\right|\right|_2^2 && (52)\\ &\leq \left|\left|UV^T\right|\right|^2 \left|\left|\mathcal{P}_{\text{row}(X)}\,\vec{x}\right|\right|_2^2 + \|W\|^2 \left|\left|\mathcal{P}_{\text{row}(X)^\perp}\,\vec{x}\right|\right|_2^2 && (53)\\ &\leq 1 \quad \text{by condition (45)}. && (54) \end{aligned}$$

Equality (51) follows from Pythagoras' Theorem because the column spaces of U and W are orthogonal by condition (46), which also implies

$$\langle W, X\rangle = 0. \tag{55}$$

By equation (191) in Lecture Notes 2

$$\left\langle UV^T, X\right\rangle = \|X\|_*\,. \tag{56}$$

For any matrix $Y \in \mathbb{R}^{m\times n}$

$$\begin{aligned} \|Y\|_* &\geq \langle G, Y\rangle \qquad \text{by (54) and Theorem 2.6 in Lecture Notes 2} && (57)\\ &= \langle G, X\rangle + \langle G, Y-X\rangle && (58)\\ &= \left\langle UV^T, X\right\rangle + \langle G, Y-X\rangle \qquad \text{by (55)} && (59)\\ &= \|X\|_* + \langle G, Y-X\rangle \qquad \text{by (56)}. && (60) \end{aligned}$$

□

2.3 Analysis of the lasso estimator

In this section we derive an exact characterization of the solution to the lasso estimator for Example 1.4. This illustrates how to use the subdifferential of a convex cost function to understand the performance of its minimizer as an estimator.

Lemma 2.11 (Sparse regression with two features)**.** *Assume that $\alpha \geq 0$ and $n \geq 2$. The lasso estimator for the sparse-regression problem in Example 1.4 is of the form*

$$\vec{\beta}_{\text{lasso}} = \begin{bmatrix} \alpha + \vec{x}_1^T \vec{z} - \lambda \\ 0 \end{bmatrix} \tag{61}$$

as long as

$$\frac{\left| \vec{x}_2^T \vec{z} - \rho \vec{x}_1^T \vec{z} \right|}{1 - |\rho|} \leq \lambda \leq \alpha + \vec{x}_1^T \vec{z}. \tag{62}$$

Proof. The lasso cost function is strictly convex if $n \geq 2$ and the matrix X is full rank (i.e. $\rho \neq 0$), because the quadratic term corresponds to a positive definite quadratic form. By Theorem 2.3, to establish that $\vec{\beta}_{\text{lasso}}$ is the unique minimizer it suffices to prove that the zero vector is a subgradient of the cost function at $\vec{\beta}_{\text{lasso}}$.

The gradient of the quadratic term

$$q\left(\vec{\beta}\right) := \frac{1}{2} \left|\left| X\vec{\beta} - \vec{y} \right|\right|_2^2 \tag{63}$$

at $\vec{\beta}_{\text{lasso}}$ equals

$$\nabla q\left(\vec{\beta}_{\text{lasso}}\right) = X^T \left(X\vec{\beta}_{\text{lasso}} - \vec{y} \right). \tag{64}$$

By Theorem 2.8, if only the first entry of $\vec{\beta}_{\text{lasso}}$ is nonzero and nonnegative, then

$$\vec{g}_{\ell_1} := \begin{bmatrix} 1 \\ \gamma \end{bmatrix} \tag{65}$$

is a subgradient of the ℓ_1 norm at $\vec{\beta}_{\text{lasso}}$ for any $\gamma \in \mathbb{R}$ such that $|\gamma| \leq 1$. By Lemmas 2.4 and 2.5, the sum of $\nabla q\left(\vec{\beta}_{\text{lasso}}\right)$ and $\lambda \vec{g}_{\ell_1}$ is a subgradient of the lasso cost function at $\vec{\beta}_{\text{lasso}}$. If only the first entry of $\vec{\beta}_{\text{lasso}}$ is nonzero, this subgradient equals

$$\begin{aligned}
\vec{g}_{\text{lasso}} &:= X^T \left(X\vec{\beta}_{\text{lasso}} - \vec{y} \right) + \lambda \begin{bmatrix} 1 \\ \gamma \end{bmatrix} && (66) \\
&= X^T \left(\vec{\beta}_{\text{lasso}}[1]\vec{x}_1 - \alpha \vec{x}_1 - \vec{z} \right) + \lambda \begin{bmatrix} 1 \\ \gamma \end{bmatrix} && (67) \\
&= \begin{bmatrix} \vec{x}_1^T \left(\vec{\beta}_{\text{lasso}}[1]\vec{x}_1 - \alpha \vec{x}_1 - \vec{z} \right) + \lambda \\ \vec{x}_2^T \left(\vec{\beta}_{\text{lasso}}[1]\vec{x}_1 - \alpha \vec{x}_1 - \vec{z} \right) + \lambda\gamma \end{bmatrix} && (68) \\
&= \begin{bmatrix} \vec{\beta}_{\text{lasso}}[1] - \alpha - \vec{x}_1^T \vec{z} + \lambda \\ \rho\vec{\beta}_{\text{lasso}}[1] - \rho\alpha - \vec{x}_2^T \vec{z} + \lambda\gamma \end{bmatrix}. && (69)
\end{aligned}$$

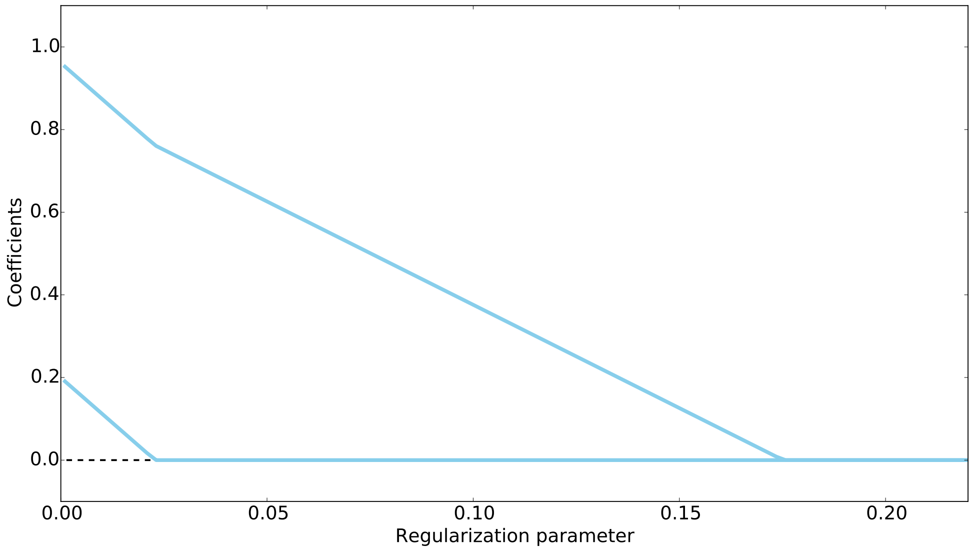

Figure 11: Coefficients of the lasso estimates in the sparse regression problem in Example 1.4 for $\alpha = 1$, 5 examples ($n = 5$), $\rho := -0.43$ and different values of the regularization parameter λ.

The expression is equal to zero if

$$\vec{\beta}_{\text{lasso}}[1] = \alpha + \vec{x}_1^T \vec{z} - \lambda, \tag{70}$$

$$\gamma = \frac{\rho\alpha + \vec{x}_2^T \vec{z} - \rho \vec{\beta}_{\text{lasso}}[1]}{\lambda} \tag{71}$$

$$= \frac{\vec{x}_2^T \vec{z} - \rho \vec{x}_1^T \vec{z}}{\lambda} + \rho. \tag{72}$$

In order to ensure that $\vec{g}_{\text{lasso}}$ is a valid subgradient for this choice, we need to check that (1) $\vec{\beta}_{\text{lasso}}[1]$ is indeed nonnegative, which is the case if λ satisfies equation (62), and (2) that $|\gamma| \leq 1$. By the triangle inequality

$$|\gamma| \leq \left| \frac{\vec{x}_2^T \vec{z} - \rho \vec{x}_1^T \vec{z}}{\lambda} \right| + |\rho| \tag{73}$$

$$\leq 1, \tag{74}$$

as long as λ satisfies equation (62). We conclude that $\vec{0}$ is a subgradient of the cost function at $\vec{\beta}_{\text{lasso}}$, which establishes that $\vec{\beta}_{\text{lasso}}$ as given by equation (61) is the unique solution to the optimization problem. □

The lemma establishes that in this example the lasso estimator detects the relevant feature vector, setting the coefficient of the irrelevant feature vector to zero, for a certain range of λ. Within that range the coefficient corresponding to the relevant predictor scales linearly with λ. This is confirmed in Figure 11.

2.4 Analysis of robust PCA

In this section we analyze the RPCA estimator showing that it succeeds for the data in Example 1.6. This is a cartoon example, but similar arguments can be used to analyze the algorithm in a more general setting [4]. The main idea is to construct a subgradient of the cost function at the ground truth that is equal to zero. This implies that the true low-rank and sparse components are a solution to the problem, but not necessarily the unique solution. The following result shows that if the subgradient satisfies two small additional constraints, then the solution is indeed unique.

Lemma 2.12 (Lemma 2.4 in [4]). *Let L^*, $S^* \in \mathbb{R}^{m\times n}$ and*

$$Y := L^* + S^*. \tag{75}$$

L^* is a rank-r matrix with SVD $U_{L^*} S_{L^*} V_{L^*}^T$, where $U_{L^*} \in \mathbb{R}^{m\times r}$, $V_{L^*} \in \mathbb{R}^{n\times r}$ and $S_{L^*} \in \mathbb{R}^{r\times r}$ contains the nonzero singular values of L^*. Assume there exists a matrix

$$G_* := U_{L^*} V_{L^*}^T + W, \tag{76}$$

where W is a matrix satisfying

$$||W|| < 1 \tag{77}$$
$$U^T W = 0, \tag{78}$$
$$W\,V = 0, \tag{79}$$

and there also exists a matrix G_{ℓ_1} satisfying

$$G_{\ell_1}[i,j] = -\operatorname{sign}\left(S^*[i,j]\right) \qquad \text{if } S^*[i,j] \neq 0, \tag{80}$$
$$|G_{\ell_1}[i,j]| < 1 \qquad \text{otherwise}, \tag{81}$$

where $S^* := Y - L^*$, such that

$$G_* + \lambda G_{\ell_1} = 0. \tag{82}$$

Then the solution to the robust PCA problem (16) is unique and equal to L^*.

Proof. By Theorem 2.10 $G_* := U_{L^*} V_{L^*}^T + W$ is a subgradient of the nuclear norm at L^*, whereas by Theorem 2.8 G_{ℓ_1} is a subgradient of $||\cdot - Y||_1$ at L^*. As a result by Lemmas 2.4 and 2.5 $G_* + \lambda G_{\ell_1}$ is a subgradient of the RPCA cost function at L^*. By Theorem 2.3 $G_* + \lambda G_{\ell_1} = 0$ consequently implies that L^* is a solution. Uniqueness follows from the strict inequalities (77) and (81). The proof is more involved and can be found in [4]. □

The following lemma establishes that the RPCA estimator recovers the low-rank and sparse components for the data in Example 1.6 for *any value* of the outlier.

Lemma 2.13. *Let*

$$Y := \begin{bmatrix} -2 & -1 & \alpha & 1 & 2 \\ -2 & -1 & 0 & 1 & 2 \\ -2 & -1 & 0 & 1 & 2 \end{bmatrix}. \tag{83}$$

For any value of α *the unique solution to the optimization problem*

$$\min_{L} ||L||_{*} + \lambda ||Y - L||_{1} \tag{84}$$

is

$$L^{*} := \begin{bmatrix} -2 & -1 & 0 & 1 & 2 \\ -2 & -1 & 0 & 1 & 2 \\ -2 & -1 & 0 & 1 & 2 \end{bmatrix} \tag{85}$$

as long as

$$\frac{2}{\sqrt{30}} < \lambda < \sqrt{\frac{2}{3}}. \tag{86}$$

Proof. In order to satisfy (80) and (76) at L^{*}, we set

$$\begin{aligned} G_{*} &= U_{L^{*}} V_{L^{*}}^{T} + W \qquad (87) \\ &= \frac{1}{\sqrt{30}} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} -2 & -1 & 0 & 1 & 2 \end{bmatrix} + W, \qquad (88) \\ G_{\ell_1}[1,3] &= -\operatorname{sign}(\alpha). \qquad (89) \end{aligned}$$

To ensure $G_{*} + \lambda G_{\ell_1} = 0$ we set

$$W := \lambda \operatorname{sign}(\alpha) \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -0.5 & 0 & 0 \\ 0 & 0 & -0.5 & 0 & 0 \end{bmatrix}, \tag{90}$$

where the entries are chosen so that (78) and (79) both hold, and

$$G_{\ell_1} = \begin{bmatrix} \frac{2}{\lambda\sqrt{30}} & \frac{1}{\lambda\sqrt{30}} & -\operatorname{sign}(\alpha) & -\frac{1}{\lambda\sqrt{30}} & -\frac{2}{\lambda\sqrt{30}} \\ \frac{2}{\lambda\sqrt{30}} & \frac{1}{\lambda\sqrt{30}} & \frac{\operatorname{sign}(\alpha)}{2} & -\frac{1}{\lambda\sqrt{30}} & -\frac{2}{\lambda\sqrt{30}} \\ \frac{2}{\lambda\sqrt{30}} & \frac{1}{\lambda\sqrt{30}} & \frac{\operatorname{sign}(\alpha)}{2} & -\frac{1}{\lambda\sqrt{30}} & -\frac{2}{\lambda\sqrt{30}} \end{bmatrix}. \tag{91}$$

To complete the proof we need to check conditions (77) and (81). We have

$$||W||^{2} = \frac{3\lambda^{2}}{2} < 1, \qquad \text{if } \lambda < \sqrt{\frac{2}{3}}, \tag{92}$$

$$|G_{\ell_1}[i,j]| \leq \max\left\{\frac{1}{2}, \frac{2}{\lambda\sqrt{30}}\right\} < 1, \qquad \text{if } \lambda > \frac{2}{\sqrt{30}}. \tag{93}$$

□

3 Minimizing nondifferentiable convex functions

3.1 Subgradient method

Consider the optimization problem

$$\text{minimize} \quad f\left(\vec{x}\right) \tag{94}$$

where f is convex but nondifferentiable. This implies that we cannot compute a gradient and advance in the steepest descent direction as in gradient descent. However, we can generalize the idea by using subgradients, which exist because f is convex. This is useful as long as it is efficient to compute the subgradient of the function.

Algorithm 3.1 (Subgradient method)**.** *We set the initial point $\vec{x}^{(0)}$ to an arbitrary value in $\mathbb{R}^n$. Then we compute*

$$\vec{x}^{(k+1)} = \vec{x}^{(k)} - \alpha_k \, \vec{g}^{(k)}, \tag{95}$$

where $\vec{g}^{(k)}$ is a subgradient of f at $\vec{x}^{(k)}$, until a convergence criterion is satisfied.

Interestingly, the subgradient method is not a descent method. The value of the cost function can actually increase as the iterations progress. However, the method can be shown to converge at a rate of order $\mathcal{O}\left(1/\epsilon^2\right)$ as long as the step size decreases along iterations, see [11].

We now apply the subgradient method to solve the lasso problem, i.e. least-squares regression with ℓ_1-norm regularization. The cost function in the optimization problem,

$$\text{minimize} \quad \frac{1}{2} \left|\left|A\vec{x} - \vec{y}\right|\right|_2^2 + \lambda \left|\left|\vec{x}\right|\right|_1, \tag{96}$$

is convex but not differentiable. By Theorem 2.8 $\operatorname{sign}\left(\vec{x}\right)$ is a subgradient of the ℓ_1 norm at $\vec{x}$, so

$$\vec{g}^{(k)} = A^T \left(A\vec{x}^{(k)} - \vec{y}\right) + \lambda \operatorname{sign}\left(\vec{x}^{(k)}\right) \tag{97}$$

is a subgradient of the cost function at $\vec{x}^{(k)}$.

Algorithm 3.2 (Subgradient method for sparse regression)**.** *Set the initial point $\vec{x}^{(0)}$ to an arbitrary value in $\mathbb{R}^n$. Update by setting*

$$\vec{x}^{(k+1)} := \vec{x}^{(k)} - \alpha_k \left(A^T \left(A\vec{x}^{(k)} - \vec{y}\right) + \lambda \operatorname{sign}\left(\vec{x}^{(k)}\right)\right), \tag{98}$$

where $\alpha_k > 0$ is the step size, until a stopping criterion is met.

Figure 12 shows the result of applying this algorithm to an example in which $A \in \mathbb{R}^{2000\times1000}$, $y = A\vec{x}^* + \vec{z}$ where $\vec{x}^*$ is 100-sparse and $\vec{z}$ is iid Gaussian. The example illustrates that decreasing the step size at each iteration achieves faster convergence.

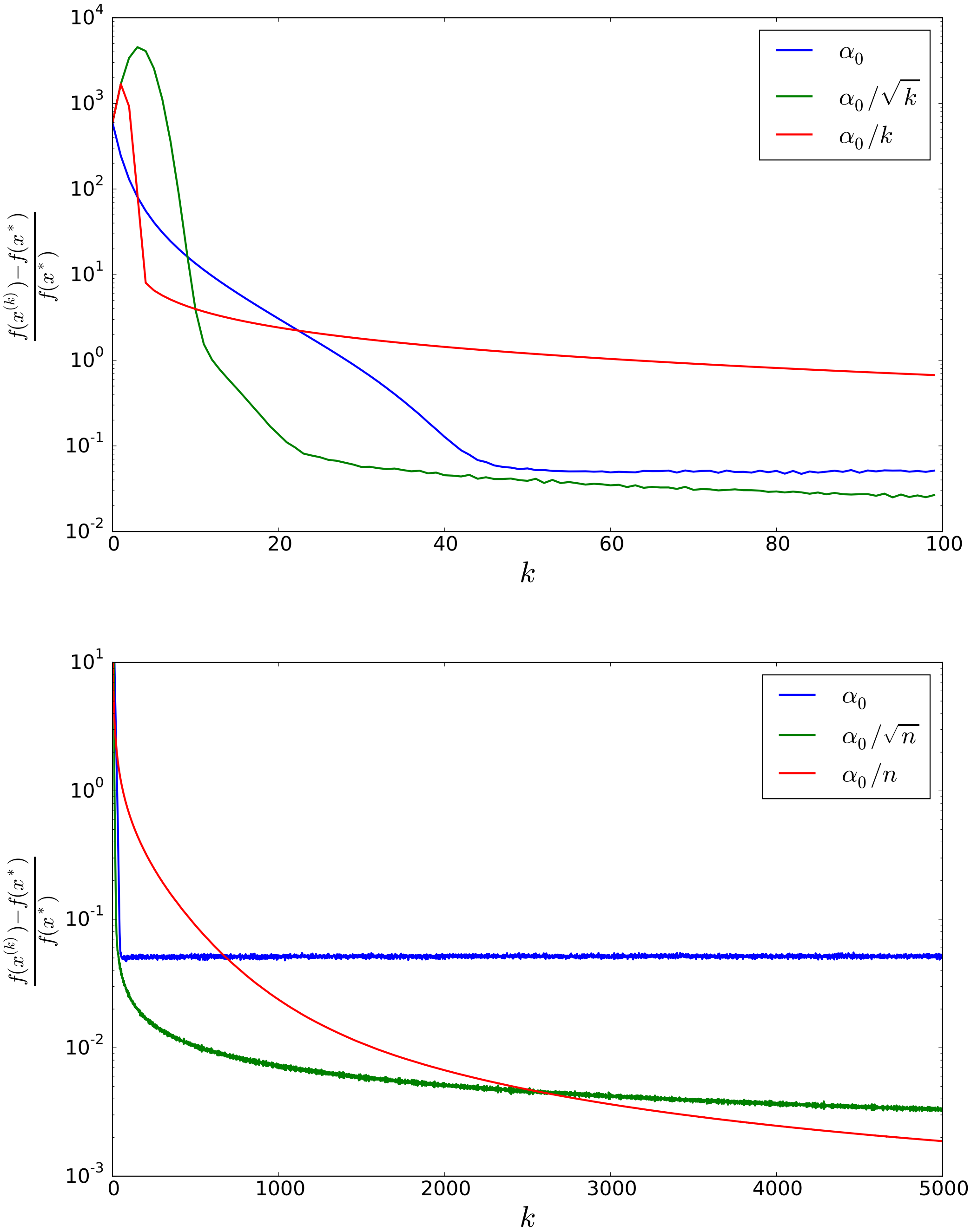

Figure 12: Subgradient method applied to least-squares regression with ℓ_1-norm regularization for different choices of step size (α_0 is a constant).

3.2 Proximal gradient method

As we saw in the previous section, convergence of subgradient method is slow, both in terms of theoretical guarantees and in the example of Figure 12. In this section we introduce an alternative method that can be applied to a class of functions which is very useful for optimization-based data analysis.

Definition 3.3 (Composite function)**.** *A composite function is a function that can be written as the sum*

$$f\left(\vec{x}\right)+h\left(\vec{x}\right) \tag{99}$$

where f convex and differentiable and h is convex but not differentiable.

Clearly, the least-squares regression cost function with ℓ_1-norm regularization is of this form.

In order to motivate proximal methods, let us begin by interpreting the gradient-descent iteration as the solution to a *local* linearization of the function.

Lemma 3.4. *The minimum of the function*

$$h\left(\vec{x}\right):=f\left(\vec{x}^{\,(k)}\right)+\nabla f\left(\vec{x}^{\,(k)}\right)^{T}\left(\vec{x}-\vec{x}^{\,(k)}\right)+\frac{1}{2\,\alpha}\left|\left|\vec{x}-\vec{x}^{\,(k)}\right|\right|_2^2 \tag{100}$$

is $\vec{x}^{\,(k)}-\alpha\nabla f\left(\vec{x}^{\,(k)}\right)$.

Proof.

$$\begin{aligned}
\vec{x}^{\,(k+1)} &:= \vec{x}^{\,(k)}-\alpha_k\,\nabla f\left(\vec{x}^{\,(k)}\right) &(101)\\
&= \arg\min_{\vec{x}}\left|\left|\vec{x}-\left(\vec{x}^{\,(k)}-\alpha_k\,\nabla f\left(\vec{x}^{\,(k)}\right)\right)\right|\right|_2^2 &(102)\\
&= \arg\min_{\vec{x}} f\left(\vec{x}^{\,(k)}\right)+\nabla f\left(\vec{x}^{\,(k)}\right)^{T}\left(\vec{x}-\vec{x}^{\,(k)}\right)+\frac{1}{2\,\alpha_k}\left|\left|\vec{x}-\vec{x}^{\,(k)}\right|\right|_2^2. &(103)
\end{aligned}$$

□

A natural generalization of gradient descent is to minimize the sum of h and the local first-order approximation of f.

$$\begin{aligned}
\vec{x}^{\,(k+1)} &= \arg\min_{\vec{x}} f\left(\vec{x}^{\,(k)}\right)+\nabla f\left(\vec{x}^{\,(k)}\right)^{T}\left(\vec{x}-\vec{x}^{\,(k)}\right)+\frac{1}{2\,\alpha_k}\left|\left|\vec{x}-\vec{x}^{\,(k)}\right|\right|_2^2+h\left(\vec{x}\right) &(104)\\
&= \arg\min_{\vec{x}}\frac{1}{2}\left|\left|x-\left(\vec{x}^{\,(k)}-\alpha_k\,\nabla f\left(\vec{x}^{\,(k)}\right)\right)\right|\right|_2^2+\alpha_k\,h\left(\vec{x}\right) &(105)\\
&= \operatorname{prox}_{\alpha_k\,h}\left(\vec{x}^{\,(k)}-\alpha_k\,\nabla f\left(\vec{x}^{\,(k)}\right)\right). &(106)
\end{aligned}$$

We have written the iteration in terms of the proximal operator of the function h.

Definition 3.5 (Proximal operator)**.** *The proximal operator of a function $h:\mathbb{R}^n\to\mathbb{R}$ is*

$$\operatorname{prox}_h\left(\vec{y}\right):=\arg\min_{\vec{x}} h\left(\vec{x}\right)+\frac{1}{2}\left|\left|\vec{x}-\vec{y}\right|\right|_2^2. \tag{107}$$

Solving the modified local first-order approximation of the composite function iteratively yields the proximal-gradient method, which will be useful if the proximal operator of h can be computed efficiently.

Algorithm 3.6 (Proximal-gradient method)**.** *We set the initial point $\vec{x}^{(0)}$ to an arbitrary value in $\mathbb{R}^n$. Then we compute*

$$\vec{x}^{(k+1)} = \operatorname{prox}_{\alpha_k h}\left(\vec{x}^{(k)} - \alpha_k \nabla f\left(\vec{x}^{(k)}\right)\right), \tag{108}$$

until a convergence criterion is satisfied.

This algorithm may be interpreted as a fixed-point method. Indeed, fixed points of the proximal-gradient iteration are a minima of the composite function and vice versa. This suggests applying the iteration repeatedly to minimize the function, although it does not prove convergence (for this we would need to prove that the operator is contractive, see [11]).

Theorem 3.7 (Fixed point of proximal operator)**.** *A vector $\vec{x}^*$ is a solution to*

$$\text{minimize} \quad f(\vec{x}) + h(\vec{x}), \tag{109}$$

if and only if it is a fixed point of the proximal-gradient iteration

$$\vec{x}^* = \operatorname{prox}_{\alpha h}\left(\vec{x}^* - \alpha \nabla f(\vec{x}^*)\right) \tag{110}$$

for any $\alpha > 0$.

Proof. $\vec{x}^*$ is a solution to the optimization problem if and only if there exists a subgradient $\vec{g}$ of h at $\vec{x}^*$ such that $\nabla f(\vec{x}^*) + \vec{g} = 0$. $\vec{x}^*$ is the solution to

$$\text{minimize} \quad \alpha\, h(\vec{x}) + \frac{1}{2}\left|\left|\vec{x}^* - \alpha \nabla f(\vec{x}^*) - x\right|\right|_2^2, \tag{111}$$

which is the case if and only if there exists a subgradient $\vec{g}$ of h at $\vec{x}^*$ such that $\alpha \nabla f(\vec{x}^*) + \alpha \vec{g} = 0$. As long as $\alpha > 0$ the two conditions are equivalent. □

Proximal methods are very useful for fitting sparse models because the proximal operator of the ℓ_1 norm is very tractable.

Theorem 3.8 (Proximal operator of ℓ_1 norm)**.** *The proximal operator of the ℓ_1 norm weighted by a constant $\alpha > 0$ is the soft-thresholding operator*

$$\operatorname{prox}_{\alpha\,||\cdot||_1}(y) = \mathcal{S}_\alpha(\vec{y}) \tag{112}$$

where

$$\mathcal{S}_\alpha(\vec{y})[i] := \begin{cases} \vec{y}[i] - \operatorname{sign}(\vec{y}[i])\,\alpha & \textit{if } |\vec{y}[i]| \geq \alpha, \\ 0 & \textit{otherwise.} \end{cases} \tag{113}$$

Proof. Writing the function as a sum,

$$\alpha \left|\left|\vec{x}\right|\right|_1 + \frac{1}{2}\left|\left|\vec{y} - \vec{x}\right|\right|_2^2 = \sum_{i=1}^{n} \alpha \left|\vec{x}[i]\right| + \frac{1}{2}\left(\vec{y}[i] - \vec{x}[i]\right)^2 \tag{114}$$

reveals that it decomposes into independent nonnegative terms. The univariate function

$$h\left(x\right) := \alpha \left|x\right| + \frac{1}{2}\left(\vec{y}[i] - x\right)^2 \tag{115}$$

is strictly convex and consequently has a unique global minimum. It is also differentiable everywhere except at zero. If $x \geq 0$ the derivative is $\lambda + x - \vec{y}[i]$, so if $\vec{y}[i] \geq \alpha$, the minimum is achieved at $\vec{y}[i] - \alpha$. If $\vec{y}[i] < \alpha$ the function is increasing for $x \geq 0$, so the minimizer must be smaller or equal to zero. The derivative for $x < 0$ is $-\alpha + x - \vec{y}[i]$ so the minimum is achieved at $\vec{y}[i] + \alpha$ if $\vec{y}[i] \leq -\alpha$. Otherwise the function is decreasing for all $x < 0$. As a result, if $-\alpha < \vec{y}[i] < \alpha$ the minimum must be at zero. □

This result yields the following algorithm for least-squares with ℓ_1-norm regularization.

Algorithm 3.9 (Iterative Shrinkage-Thresholding Algorithm (ISTA))**.** *We set the initial point* $\vec{x}^{\,(0)}$ *to an arbitrary value in* $\mathbb{R}^n$. *Then we compute*

$$\vec{x}^{\,(k+1)} = \mathcal{S}_{\alpha_k \lambda}\left(\vec{x}^{\,(k)} - \alpha_k\, A^T\left(A\vec{x}^{\,(k)} - \vec{y}\right)\right), \tag{116}$$

until a convergence criterion is satisfied.

ISTA can be accelerated using a momentum term as in Nesterov's accelerated gradient method. This yields a fast version of the algorithm called FISTA.

Algorithm 3.10 (Fast Iterative Shrinkage-Thresholding Algorithm (FISTA))**.** *We set the initial point* $\vec{x}^{\,(0)}$ *to an arbitrary value in* $\mathbb{R}^n$. *Then we compute*

$$\vec{z}^{\,(0)} = \vec{x}^{\,(0)} \tag{117}$$

$$\vec{x}^{\,(k+1)} = \mathcal{S}_{\alpha_k \lambda}\left(\vec{z}^{\,(k)} - \alpha_k\, A^T\left(A\vec{z}^{\,(k)} - \vec{y}\right)\right), \tag{118}$$

$$\vec{z}^{\,(k+1)} = \vec{x}^{\,(k+1)} + \frac{k}{k+3}\left(\vec{x}^{\,(k+1)} - \vec{x}^{\,(k)}\right), \tag{119}$$

until a convergence criterion is satisfied.

ISTA and FISTA were proposed by Beck and Teboulle in [1]. ISTA is a descent method. It has the same convergence rate as gradient descent $\mathcal{O}\left(1/\epsilon\right)$ both with a constant step size and with a backtracking line search, under the condition that ∇f be L-Lipschitz continuous. FISTA in contrast is not a descent method, but it can be shown to converge in $\mathcal{O}\left(1/\sqrt{\epsilon}\right)$ to an ϵ-optimal solution.

To illustrate the performance of ISTA and FISTA, we apply them to the same example used in Figure 12. Even without applying a backtracking line search both methods converge to a solution of middle precision (around 10^{-3} or 10^{-4}) much more rapidly than the subgradient method. The results are shown in Figure 13.

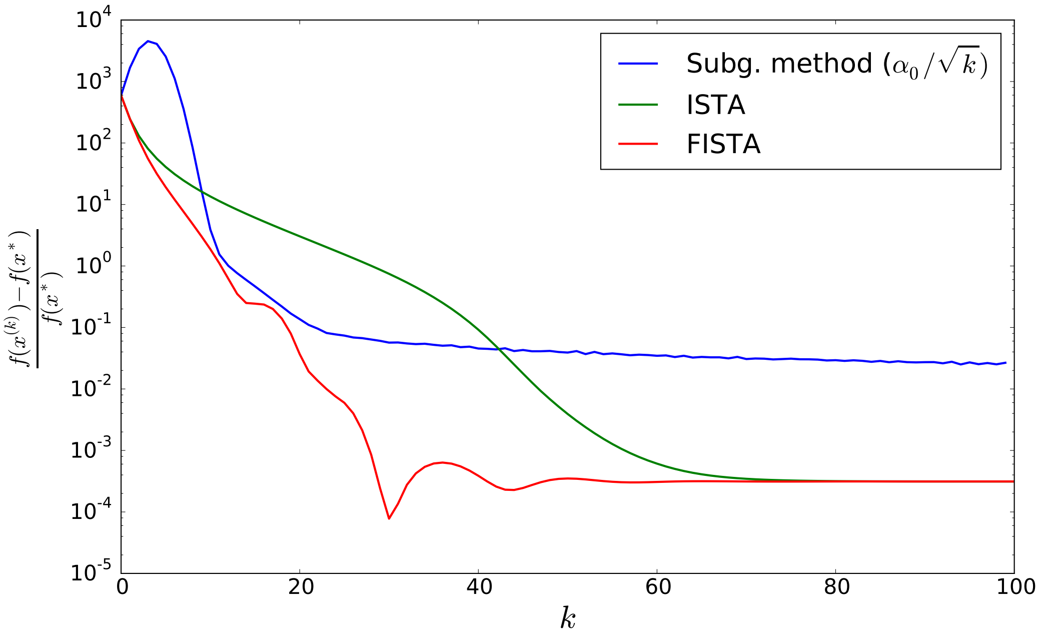

Figure 13: ISTA and FISTA applied to least-squares regression with ℓ_1-norm regularization.

4 Proofs

4.1 Proof of Theorem 2.2

We prove the statement about convexity. The statement about strict convexity can be proved in a similar way.

The epigraph of a convex function is a convex set, meaning that it contains the line between any of its points. As a consequence of the separating-hyperplane theorem, which states that there is a separating hyperplane between any two disjoint convex sets (we omit the proof which can be found in any text on convex analysis), such sets have a supporting hyperplane at every point. This establishes that convex functions defined on $\mathbb{R}^n$ have a subgradient at every point.

Now assume that a function has a subgradient at every point. The for any $\vec{x}, \vec{y} \in \mathbb{R}^n$ and $\alpha \in \mathbb{R}$ there exists a subgradient $\vec{g}$ of f at $\alpha\vec{x} + (1-\alpha)\vec{y}$. This implies

$$f(\vec{y}) \geq f(\alpha\vec{x} + (1-\alpha)\vec{y}) + \vec{g}^{\,T}(y - \alpha\vec{x} - (1-\alpha)\vec{y}) \tag{120}$$

$$= f(\alpha\vec{x} + (1-\alpha)\vec{y}) + \alpha\,\vec{g}^{\,T}(y - \vec{x}), \tag{121}$$

$$f(\vec{x}) \geq f(\alpha\vec{x} + (1-\alpha)\vec{y}) + \vec{g}^{\,T}(\vec{x} - \alpha\vec{x} - (1-\alpha)\vec{y}) \tag{122}$$

$$= f(\alpha\vec{x} + (1-\alpha)\vec{y}) + (1-\alpha)\,\vec{g}^{\,T}(y - \vec{x}). \tag{123}$$

Multiplying equation (121) by $1-\alpha$ and equation (123) by α and adding them together yields

$$\alpha f(\vec{x}) + (1-\alpha) f(\vec{y}) \geq f(\alpha\vec{x} + (1-\alpha)\vec{y}). \tag{124}$$

We conclude that the function is convex.

References

A very readable and exhaustive reference on convex optimization is Boyd and Vandenberghe's seminal book [2], which unfortunately does not cover subgradients.[2] Nesterov's book [7] and Rockafellar's book [8] do cover subgradients. Chapter 5 of [6] in Hastie, Tibshirani and Wainwright is a great description of proximal-gradient methods, as well as alternative first-order techniques such as coordinate descent, and their application to sparse regression.

[1] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

[2] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[3] E. J. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):11, 2011.

[4] E. J. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):11, 2011.

[5] V. Chandrasekaran, S. Sanghavi, P. A. Parrilo, and A. S. Willsky. Rank-sparsity incoherence for matrix decomposition. *SIAM Journal on Optimization*, 21(2):572–596, 2011.

[6] T. Hastie, R. Tibshirani, and M. Wainwright. *Statistical learning with sparsity: the lasso and generalizations*. CRC Press, 2015.

[7] Y. Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Springer Publishing Company, Incorporated, 1 edition, 2014.

[8] R. T. Rockafellar and R. J.-B. Wets. *Variational analysis*. Springer Science & Business Media, 2009.

[9] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[10] P. Tseng. Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of optimization theory and applications*, 109(3):475–494, 2001.

[11] L. Vandenberghe. Notes on optimization methods for large-scale systems.

[12] G. A. Watson. Characterization of the subdifferential of some matrix norms. *Linear algebra and its applications*, 170:33–45, 1992.

[2] However, see `http://see.stanford.edu/materials/lsocoee364b/01-subgradients_notes.pdf`