

Low-rank models

1 Matrix completion

1.1 The matrix-completion problem

The Netflix Prize was a contest organized by Netflix from 2007 to 2009 in which teams of data scientists tried to develop algorithms to improve the prediction of movie ratings. The problem of predicting ratings can be recast as that of completing a matrix from some of its entries, as illustrated in Figure 1. This problem is known as *matrix completion*.

At first glance, the problem of completing a matrix such as this one

$$\begin{bmatrix} 1 & ? & 5 \\ ? & 3 & 2 \end{bmatrix} \quad (1)$$

may seem completely ill posed. We can just fill in the missing entries arbitrarily! In more mathematical (and perhaps slightly pedantic) terms, the completion problem is equivalent to an underdetermined system of equations

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} M_{11} \\ M_{21} \\ M_{12} \\ M_{22} \\ M_{13} \\ M_{23} \end{bmatrix} = \begin{bmatrix} 1 \\ 3 \\ 5 \\ 2 \end{bmatrix}. \quad (2)$$

In order to solve the problem, we need to make an assumption on the structure of the matrix that we aim to complete. Recall that in compressed sensing we made the assumption that the original signal was sparse. Even though the recovery problem in compressed sensing is also underdetermined, recovery is possible as long as the number of measurements is proportional to the number of nonzero entries (up to logarithmic factors). In the case of matrix completion, we will make the assumption that the original matrix is low rank. This implies that there exists a high correlation between the entries of the matrix, which may make it possible to infer the missing entries from the observations. As a very simple example,



Figure 1: A depiction of the Netflix challenge in matrix form. Each row corresponds to a user that ranks a subset of the movies, which correspond to the columns. The figure is due to Mahdi Soltanolkotabi.

consider the following matrix

$$\begin{bmatrix} 1 & 1 & 1 & 1 & ? & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ ? & 1 & 1 & 1 & 1 & 1 \end{bmatrix}. \quad (3)$$

Setting the missing entries to 1 yields a rank 1 matrix, whereas setting them to any other number yields a rank 2 or rank 3 matrix.

The low-rank assumption implies that if the matrix has dimensions $m \times n$ then it can be factorized into two matrices that have dimensions $m \times r$ and $r \times n$. This factorization allows to encode the matrix using $r(m + n)$ parameters. If the number of observed entries is larger than $r(m + n)$ parameters then it may be possible to recover the missing entries. However, this is not enough to ensure that the problem is well posed, as we will see in the following section.

1.2 When does matrix completion make sense?

The results of matrix completion will obviously depend on the subset of entries that are observed. For example, completion is impossible unless we observe at least one entry in each

row and column. To see why let us consider a rank 1 matrix for which we don't observe the second row,

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ ? & ? & ? & ? \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 \\ ? \\ 1 \\ 1 \end{bmatrix} [1 \ 1 \ 1 \ 1]. \quad (4)$$

As long as we set the missing row to equal the same value, we will obtain a rank-1 matrix consistent with the measurements. In this case, the problem is not well posed.

In general, we need samples that are distributed across the whole matrix. This may be achieved by sampling entries uniformly at random. Although this model does not completely describe matrix completion problems in practice (some users tend to rate more movies, some movies are very popular and are rated by many people), making the assumption that the revealed entries are random simplifies theoretical analysis and avoids dealing with adversarial cases designed to make deterministic patterns fail.

We now turn to the question of what matrices can be completed from a subset of entries samples uniformly at random. Intuitively, matrix completion can be achieved when the information contained in the entries of the matrix is *spread out* across multiple entries. If the information is very localized then it will be impossible to reconstruct the missing entries. Consider a simple example where the matrix is sparse

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 23 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \quad (5)$$

If we don't observe the nonzero entry, we will naturally assume that it was equal to zero.

The problem is not restricted to sparse matrices. In the following matrix the last row does not seem to be correlated to the rest of the rows,

$$M := \begin{bmatrix} 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 \\ -3 & 3 & -3 & 3 \end{bmatrix}. \quad (6)$$

This is revealed by the singular-value decomposition of the matrix, which allows to decompose

it into two rank-1 matrices.

$$M = U \Sigma V^T \tag{7}$$

$$= \begin{bmatrix} 0.5 & 0 \\ 0.5 & 0 \\ 0.5 & 0 \\ 0.5 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 8 & 0 \\ 0 & 6 \end{bmatrix} \begin{bmatrix} 0.5 & 0.5 & 0.5 & 0.5 \\ -0.5 & 0.5 & -0.5 & 0.5 \end{bmatrix} \tag{8}$$

$$= 8 \begin{bmatrix} 0.5 \\ 0.5 \\ 0.5 \\ 0.5 \\ 0 \end{bmatrix} \begin{bmatrix} 0.5 & 0.5 & 0.5 & 0.5 \end{bmatrix} + 6 \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} -0.5 & 0.5 & -0.5 & 0.5 \end{bmatrix} \tag{9}$$

$$= \sigma_1 U_1 V_1^T + \sigma_2 U_2 V_2^T. \tag{10}$$

The first rank-1 component of this decomposition has information that is very spread out,

$$\sigma_1 U_1 V_1^T = \begin{bmatrix} 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \tag{11}$$

The reason is that most of the entries of V_1 are nonzero and have the same magnitude, so that each entry of U_1 affects every single entry of the corresponding row. If one of those entries is missing, we can still recover the information from the other entries.

In contrast, the information in the second rank-1 component is very localized, due to the fact that the corresponding left singular vector is very sparse,

$$\sigma_2 U_2 V_2^T = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -3 & 3 & -3 & 3 \end{bmatrix}. \tag{12}$$

Each entry of the right singular vector only affects one entry of the component. If we don't observe that entry then it will be impossible to recover.

This simple example shows that sparse singular vectors are problematic for matrix completion. In order to quantify to what extent the information is spread out across the low-rank matrix we define a coherence measure that depends on the singular vectors.

Definition 1.1 (Coherence). *Let $U \Sigma V^T$ be the singular-value decomposition of an $n \times n$ matrix M with rank r . The coherence μ of M is a constant such that*

$$\max_{1 \leq j \leq n} \sum_{i=1}^r U_{ij}^2 \leq \frac{n\mu}{r} \quad (13)$$

$$\max_{1 \leq j \leq n} \sum_{i=1}^r V_{ij}^2 \leq \frac{n\mu}{r}. \quad (14)$$

This condition was first introduced in [3]. Its exact formulation is not too important. The point is that matrix completion from uniform samples only makes sense for matrices which are incoherent, and therefore do not have spiky singular values. There is a direct analogy with the super-resolution problem, where sparsity is not a strong enough constraint to make the problem well posed and the class of signals of interest has to be further restricted to signals with supports that satisfy a minimum separation.

1.3 The nuclear norm

In compressed sensing and super-resolution we penalize the ℓ_1 norm of the recovered signal to promote sparse estimates. Similarly, for matrix completion we penalize a certain matrix norm to promote low-rank structure.

First, let us introduce the trace operator and an inner product for matrices.

Definition 1.2 (Trace). *The trace of an $n \times n$ matrix is defined as*

$$\text{trace}(M) := \sum_{i=1}^n M_{ii}. \quad (15)$$

Definition 1.3 (Matrix inner product). *The inner product between two $m \times n$ matrices A and B is*

$$\langle A, B \rangle := \text{trace}(A^T B) \quad (16)$$

$$= \sum_{i=1}^m \sum_{j=1}^n A_{ij} B_{ij}. \quad (17)$$

Note that this inner product is equivalent to the inner product of the matrices if we vectorize them.

We now define three matrix norms.

Definition 1.4 (Matrix norms). Let $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n$ be the singular values of $M \in \mathbb{R}^{m \times n}$, $m \geq n$. The operator norm is equal to the maximum singular value

$$\|M\| := \max_{\|u\|_2 \leq 1} \|Mu\|_2 \quad (18)$$

$$= \sigma_1. \quad (19)$$

The Frobenius norm is the norm induced by the inner product from Definition 1.3. It is equal to the ℓ_2 norm of the vectorized matrix, or equivalently to the ℓ_2 norm of the singular values

$$\|M\|_F := \sqrt{\sum_i M_{ij}^2} \quad (20)$$

$$= \sqrt{\text{trace}(M^T M)} \quad (21)$$

$$= \sqrt{\sum_{i=1}^n \sigma_i^2}. \quad (22)$$

The nuclear norm is equal to the ℓ_1 norm of the singular values

$$\|M\|_* := \sum_{i=1}^n \sigma_i. \quad (23)$$

The following proposition, proved in Section A.1 of the appendix, is analogous to Hölder's inequality for vectors.

Proposition 1.5. For any matrix $A \in \mathbb{R}^{m \times n}$,

$$\|A\|_* = \sup_{\|B\| \leq 1} \langle A, B \rangle. \quad (24)$$

A direct consequence of the proposition is that the nuclear norm satisfies the triangle inequality. This implies that it is a norm (it clearly satisfies the other properties of a norm) and hence a convex function.

Corollary 1.6. For any $m \times n$ matrices A and B

$$\|A + B\|_* \leq \|A\|_* + \|B\|_*. \quad (25)$$

Proof.

$$\|A + B\|_* = \sup_{\{C \mid \|C\| \leq 1\}} \langle A + B, C \rangle \quad (26)$$

$$= \sup_{\|C\| \leq 1} \langle A, C \rangle + \sup_{\|D\| \leq 1} \langle B, D \rangle \quad (27)$$

$$= \|A\|_* + \|B\|_*. \quad (28)$$

□

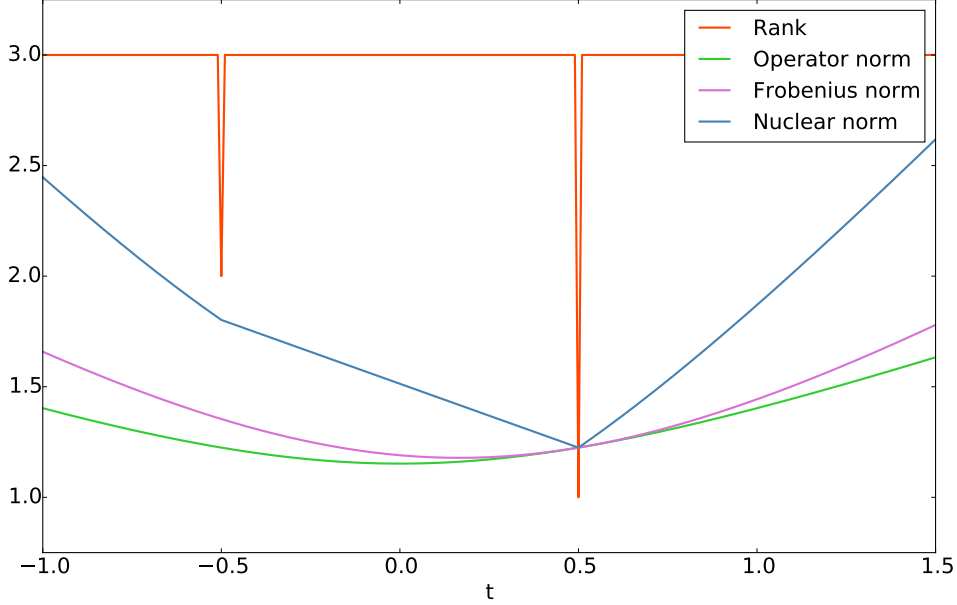


Figure 2: Values of different norms for the matrix $M(t)$ defined by (29). The rank of the matrix for each t is marked in orange.

Penalizing the nuclear norm induces low-rank structure, just like penalizing the ℓ_1 norm of a vector induces sparse structure. A justification for this is that the nuclear norm is the ℓ_1 norm of the singular values and therefore minimizing it tends to set most of them to zero. In order to provide a more concrete example, let us consider the following matrix

$$M(t) := \begin{bmatrix} 0.5 + t & 1 & 1 \\ 0.5 & 0.5 & t \\ 0.5 & 1 - t & 0.5 \end{bmatrix}, \quad (29)$$

which is parametrized by the parameter t . In Figure 2 we compare the rank, the operator norm, the Frobenius norm and the nuclear norm of $M(t)$ for different values of t . The value of t that minimizes the rank is the same as the one that minimizes the nuclear norm. In contrast, the values of t that minimize the operator and Frobenius norms are different. This justifies using the nuclear norm as a proxy for the rank.

As we discussed in the previous sections, we are interested in recovering low-rank matrices from a subset of their entries. Let y be a vector containing the revealed entries and let Ω be the corresponding entries. Ideally, we would like to select the matrix with the lowest rank that corresponds to the measurements,

$$\min_{\tilde{X} \in \mathbb{R}^{m \times n}} \text{rank}(\tilde{X}) \quad \text{such that } \tilde{X}_\Omega = y. \quad (30)$$

Unfortunately, this optimization problem is computationally hard to solve. Substituting the rank with the nuclear norm yields a tractable alternative:

$$\min_{\tilde{X} \in \mathbb{R}^{m \times n}} \left\| \tilde{X} \right\|_* \quad \text{such that } \tilde{X}_\Omega = y. \quad (31)$$

The cost function is convex and the constraint is linear, so this is a convex program. In practice, the revealed entries are usually noisy. They do not correspond exactly to entries from a low-rank matrix. We take this into account by removing the equality constraint and adding a data-fidelity term penalizing the ℓ_2 -norm error over the revealed entries in the cost function,

$$\min_{\tilde{X} \in \mathbb{R}^{m \times n}} \frac{1}{2} \left\| \tilde{X}_\Omega - y \right\|_2^2 + \lambda \left\| \tilde{X} \right\|_*, \quad (32)$$

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

We now apply this method to the following completion problem:

$$\begin{array}{cccc} \text{Bob} & \text{Molly} & \text{Mary} & \text{Larry} \\ \left(\begin{array}{cccc} 1 & ? & 5 & 4 \\ ? & 1 & 4 & 5 \\ 4 & 5 & 2 & ? \\ 5 & 4 & 2 & 1 \\ 4 & 5 & 1 & 2 \\ 1 & 2 & ? & 5 \end{array} \right) & \begin{array}{l} \text{The Dark Knight} \\ \text{Spiderman 3} \\ \text{Love Actually} \\ \text{Bridget Jones's Diary} \\ \text{Pretty Woman} \\ \text{Superman 2} \end{array} \end{array} \quad (33)$$

In more detail we apply the following steps:

1. We compute the average observed rating and subtract it from each entry in the matrix. We denote the vector of centered ratings by y .
2. We solve the optimization problem (31).
3. We add the average observed rating to the solution of the optimization problem and round each entry to the nearest integer.

The result is pretty good,

$$\begin{array}{cccc} \text{Bob} & \text{Molly} & \text{Mary} & \text{Larry} \\ \left(\begin{array}{cccc} 1 & \mathbf{2} (1) & 5 & 4 \\ \mathbf{2} (2) & 1 & 4 & 5 \\ 4 & 5 & 2 & \mathbf{2} (1) \\ 5 & 4 & 2 & 1 \\ 4 & 5 & 1 & 2 \\ 1 & 2 & \mathbf{5} (5) & 5 \end{array} \right) & \begin{array}{l} \text{The Dark Knight} \\ \text{Spiderman 3} \\ \text{Love Actually} \\ \text{Bridget Jones's Diary} \\ \text{Pretty Woman} \\ \text{Superman 2} \end{array} \end{array} \quad (34)$$

For comparison the original ratings are shown in brackets.

1.4 Theoretical guarantees

In this section we will explain how to establish theoretical guarantees for matrix completion via nuclear norm minimization. The following theorem is proved in [6].

Theorem 1.7 (Matrix completion). *If the matrix M has rank r and coherence μ , the solution to the optimization problem*

$$\min_{\tilde{X} \in \mathbb{R}^{m \times n}} \left\| \tilde{X} \right\|_* \quad \text{such that } \tilde{X}_\Omega = y \quad (35)$$

achieves exact recovery with high probability as long as the number of samples is proportional to $\mu r (n + m)$ (up to logarithmic terms).

For matrices that are incoherent, the coherence parameter is equal to a constant, so that nuclear norm allows to reconstruct the matrix from a minimal number of measurements (recall that a low-rank matrix depends on $r(n + m)$ parameters), up to logarithmic factors¹.

In order to prove that a certain matrix is the solution to the optimization problem we use the same technique that we used to analyze compressed sensing and super-resolution: we build a dual certificate. The dual certificate is a subgradient with a certain structure, so we first define the subgradients of the nuclear norm.

Proposition 1.8 (Subgradients of the nuclear norm). *Let $M = U\Sigma V^T$ be the singular-value decomposition of M . Any matrix of the form*

$$G := UV^T + W \quad \|W\| \leq 1, \quad (36)$$

$$U^T W = 0, \quad (37)$$

$$WV = 0 \quad (38)$$

is a subgradient of the nuclear norm at M .

Proof. Note that by definition $\|G\| \leq 1$. For any matrix H

$$\|M + H\|_* \geq \langle G, M + H \rangle \quad \text{by Proposition 1.5} \quad (39)$$

$$= \langle UV^T, M \rangle + \langle G, H \rangle \quad (40)$$

$$= \|M\|_* + \langle G, H \rangle \quad \text{as in (137)}. \quad (41)$$

□

The following proposition provides a dual certificate for the nuclear-norm minimization problem.

¹In fact, one can show that a multiplicative logarithmic factor is necessary in order to make sure that all rows and columns are sampled. See [4].

Proposition 1.9 (Dual certificate for nuclear-norm minimization). *Let $M = U\Sigma V^T$ be the singular-value decomposition of M . A matrix Q supported on Ω , i.e. such that*

$$Q_{\Omega^c} = 0, \tag{42}$$

is a dual certificate of the optimization problem

$$\min_{\tilde{X} \in \mathbb{R}^{m \times n}} \|\tilde{X}\|_* \quad \text{such that } \tilde{X}_\Omega = y \tag{43}$$

as long as

$$Q = UV^T + W, \quad \|W\| < 1, \tag{44}$$

$$U^T W = 0, \tag{45}$$

$$WV = 0. \tag{46}$$

Proof. By Proposition 1.8 Q is a subgradient of the nuclear norm at M . Any matrix that is feasible for the optimization problem can be expressed as $M + H$ where $H_\Omega = 0$ because the revealed entries must be equal to M_Ω . This immediately implies that $\langle Q, H \rangle = 0$. We conclude that

$$\|M + H\|_* \geq \|M\|_* + \langle Q, H \rangle \tag{47}$$

$$= \|M\|_*. \tag{48}$$

This proves that M is a solution. A variation of this argument that uses the strict inequality in (44) establishes that if Q exists then M is the unique solution [3]. \square

In order to show that matrix completion via nuclear-norm minimization succeeds, we need to show that such a dual certificate exists with high probability. For this we will need the matrix to be incoherent, since otherwise UV^T may have large entries which are not in Ω . This would make it very challenging to construct Q in a way that $UV^T = Q - W$ for a matrix W with bounded operator norm. The first guarantees for matrix completion were obtained by constructing such a certificate in [3] and [4]. Subsequently, the results were improved in [6], where it is shown that an *approximate* dual certificate also allows to establish exact recovery, and simplifies the proofs significantly.

1.5 Algorithms

In this section we describe a proximal-gradient method to solve Problem 32. Recall that proximal-gradient methods allow to solve problems of the form

$$\text{minimize } f(x) + g(x), \tag{49}$$

where f is differentiable and we can apply the proximal operator prox_g efficiently.

Recall that the proximal operator norm of the ℓ_1 norm is a soft-thresholding operator. Analogously, the proximal operator of the nuclear norm is applied by soft-thresholding the singular values of the matrix. The result is proved in Section 1.10 of the appendix.

Proposition 1.10 (Proximal operator of the nuclear norm). *The solution to*

$$\min_{\tilde{X} \in \mathbb{R}^{m \times n}} \frac{1}{2} \left\| Y - \tilde{X} \right\|_{\text{F}}^2 + \tau \left\| \tilde{X} \right\|_* \quad (50)$$

is $\mathcal{D}_\tau(Y)$, obtained by soft-thresholding the singular values of $Y = U \Sigma V^T$

$$\mathcal{D}_\tau(Y) := U \mathcal{S}_\tau(\Sigma) V^T, \quad (51)$$

$$\mathcal{S}_\tau(\Sigma)_{ii} := \begin{cases} \Sigma_{ii} - \tau & \text{if } \Sigma_{ii} > \tau, \\ 0 & \text{otherwise.} \end{cases} \quad (52)$$

Algorithm 1.11 (Proximal-gradient method for nuclear-norm regularization). *Let Y be a matrix such that $Y_\Omega = y$ and let us abuse notation by interpreting $X_\Omega^{(k)}$ as a matrix which is zero on Ω^c . We set the initial point $X^{(0)}$ to Y . Then we iterate the update*

$$X^{(k+1)} = \mathcal{D}_{\alpha_k \lambda} \left(X^{(k)} - \alpha_k \left(X_\Omega^{(k)} - Y \right) \right), \quad (53)$$

where $\alpha_k > 0$ is the step size.

1.6 Alternating minimization

Minimizing the nuclear norm to recover a low-rank matrix is an effective method but it has a drawback: it requires repeatedly computing the singular-value decomposition of the matrix, which can be computationally heavy for large matrices. A more efficient alternative is to parametrize the matrix as AB where $A \in \mathbb{R}^{m \times k}$ and $B \in \mathbb{R}^{k \times n}$, which requires fixing a value for the rank of the matrix k (in practice this can be set by cross validation). The two components A and B can then be fit by solving the optimization problem

$$\min_{\tilde{A} \in \mathbb{R}^{m \times k}, \tilde{B} \in \mathbb{R}^{k \times n}} \left\| \left(\tilde{A} \tilde{B} \right)_\Omega - y \right\|_2. \quad (54)$$

This nonconvex problem is usually tackled by alternating minimization. Indeed, if we fix $\tilde{B} = B$ the optimization problem over \tilde{A} is just a least-squares problem

$$\min_{\tilde{A} \in \mathbb{R}^{m \times k}} \left\| \left(\tilde{A} B \right)_\Omega - y \right\|_2 \quad (55)$$

and the same is true for the optimization problem over \tilde{B} if we fix $\tilde{A} = A$. Iteratively solving these least-squares problems allows to find a local minimum of the cost function. Under certain assumptions, one can even show that a certain initialization coupled with this procedure guarantees exact recovery, see [8] for more details.

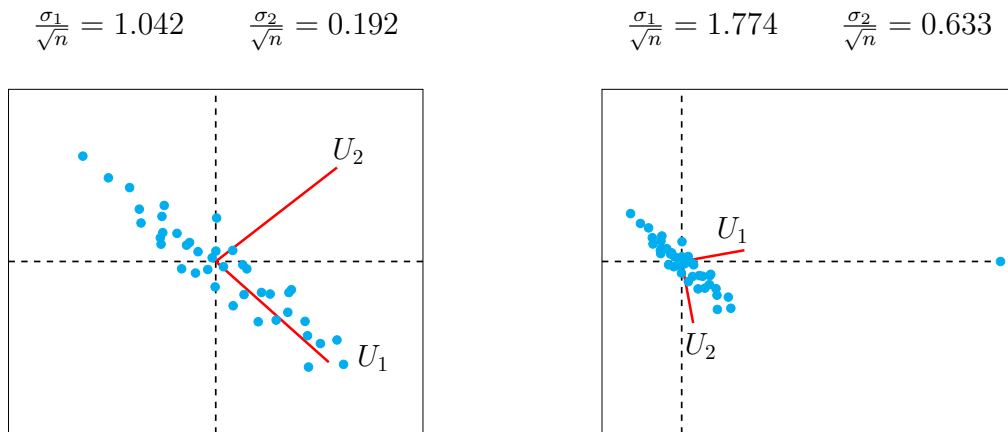


Figure 3: Principal components of a dataset before and after the inclusion of an outlier.

2 Low rank + sparse model

2.1 The effect of outliers on principal-component analysis

It is well known that outliers may severely distort the results of applying principal-component analysis to a dataset. Figure 3 shows the dramatic effect that just one outlier can have on the principal components. Equivalently, if several entries of a matrix are entirely uncorrelated with the rest, this may disrupt any low-rank structure that might be present. To illustrate this consider a rank-1 matrix of movie ratings

$$\begin{array}{cccc}
 & \text{Bob} & \text{Molly} & \text{Mary} & \text{Larry} & \\
 \left(\begin{array}{cccc}
 1 & 1 & 5 & 5 \\
 1 & 1 & 5 & 5 \\
 5 & 5 & 1 & 1 \\
 5 & 5 & 1 & 1 \\
 5 & 5 & 1 & 1 \\
 1 & 1 & 5 & 5
 \end{array} \right) & \begin{array}{l}
 \text{The Dark Knight} \\
 \text{Spiderman 3} \\
 \text{Love Actually} \\
 \text{Bridget Jones's Diary} \\
 \text{Pretty Woman} \\
 \text{Superman 2}
 \end{array} & (56)
 \end{array}$$

Now imagine that Bob randomly assigns a 5 instead of a 1 to The Dark Knight by mistake and that Larry hates Superman 2 because one of the actresses reminds him of his ex,

$$\begin{matrix}
 & \text{Bob} & \text{Molly} & \text{Mary} & \text{Larry} & & \\
 \left(\begin{matrix}
 \color{red}{5} & 1 & 5 & 5 \\
 1 & 1 & 5 & 5 \\
 5 & 5 & 1 & 1 \\
 5 & 5 & 1 & 1 \\
 5 & 5 & 1 & 1 \\
 1 & 1 & 5 & \color{red}{1}
 \end{matrix} \right) & \begin{matrix}
 \text{The Dark Knight} \\
 \text{Spiderman 3} \\
 \text{Love Actually} \\
 \text{Bridget Jones's Diary} \\
 \text{Pretty Woman} \\
 \text{Superman 2}
 \end{matrix} & & (57)
 \end{matrix}$$

Now let us compare the singular-value decomposition after subtracting the mean rating with

$$U\Sigma V^T = U \begin{bmatrix} 8.543 & 0 & 0 & 0 \\ 0 & 4.000 & 0 & 0 \\ 0 & 0 & 2.649 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} V^T \quad (58)$$

and without outliers

$$A - \bar{A} = U\Sigma V^T = U \begin{bmatrix} 9.798 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} V^T. \quad (59)$$

The matrix is now rank 3 instead of rank 1. In addition, the first left singular vector,

$$U_1 = \begin{matrix} \text{D. Knight} & \text{Sp. 3} & \text{Love Act.} & \text{B.J.'s Diary} & \text{P. Woman} & \text{Sup. 2} \\ (-0.2610 & -0.4647 & 0.4647 & 0.4647 & 0.4647 & -0.2610) \end{matrix} \quad (60)$$

does not allow to cluster the movies as effectively as when there are no outliers

$$U_1 = \begin{matrix} \text{D. Knight} & \text{Sp. 3} & \text{Love Act.} & \text{B.J.'s Diary} & \text{P. Woman} & \text{Sup. 2} \\ (-0.4082 & -0.4082 & 0.4082 & 0.4082 & 0.4082 & -0.4082) \end{matrix}. \quad (61)$$

2.2 Low rank + sparse model

In order to tackle situations where a low-rank matrix may be corrupted with outliers, we define a low-rank + sparse model where the matrix is modeled as the sum of a low-rank component L and a sparse component S . Figure 4 shows an example with simulated data. If we are able to separate the two components from the data, we can apply PCA to L in order

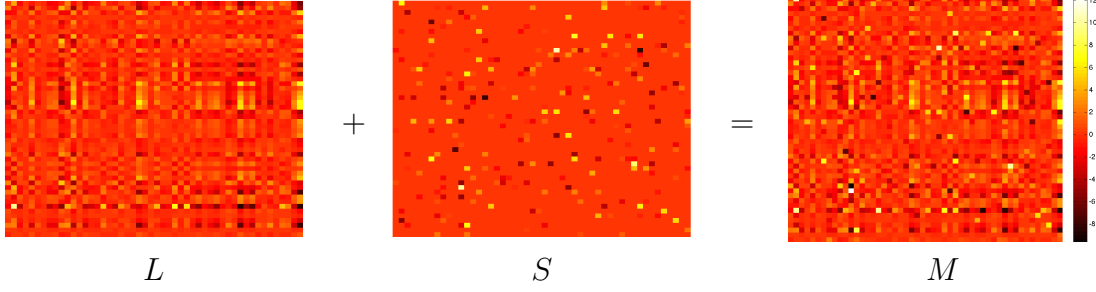


Figure 4: M is obtained by summing a low-rank matrix L and a sparse matrix S .

to retrieve the low-rank structure that is distorted by the presence of S . The problem of separating both components is very related to matrix completion: if we knew the location of the outliers then we could just apply a matrix-completion method to the remaining entries in order to recover the low-rank component.

We now consider under what assumptions the decomposition of a low rank and a sparse component is unique. Clearly the low-rank component cannot be sparse, otherwise there can be many possible decompositions

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 23 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 47 & 0 \end{bmatrix} \quad (62)$$

$$= \quad (63)$$

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 23 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 47 & 0 \end{bmatrix}. \quad (64)$$

We can avoid this by considering low-rank components with low coherence.

Similarly, the sparse component cannot be low rank. This occurs when its support is highly concentrated on a small number of columns or rows,

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \quad (65)$$

$$= \quad (66)$$

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 & 2 & 2 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \quad (67)$$

A simple assumption that precludes the support of the sparse component from being too concentrated is that it is distributed uniformly at random. As in the case of the revealed entries in matrix completion, this assumption doesn't usually hold in practice but it often provides a good approximation.

2.3 Robust principal-component analysis

Following the ideas that we discussed in previous lecture notes and in Section 1.3, we compute the decomposition by penalizing the nuclear norm of L , which induces low-rank structure, and the ℓ_1 norm of S , which induces sparse structure.

$$\min_{\tilde{L}, \tilde{S} \in \mathbb{R}^{m \times n}} \left\| \tilde{L} \right\|_* + \lambda \left\| \tilde{S} \right\|_1 \quad \text{such that } \tilde{L} + \tilde{S} = M, \quad (68)$$

where M is the matrix of data. To be clear, $\|\cdot\|_1$ is the ℓ_1 norm of the vectorized matrix and $\lambda > 0$ is a regularization parameter. This algorithm introduced by [2,5] is often called *robust PCA*, since it aims to obtain a low-rank component that is not affected by the presence of outliers.

The regularization parameter λ determines the weight of the two structure-inducing norms. It may be chosen by cross validation, but its possible values may be restricted by considering some simple examples. If the matrix that we are decomposing is equal to a rank-1 matrix, then the sparse component should be set to zero. Since

$$\left\| \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} \right\|_* = n, \quad \left\| \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} \right\|_1 = n^2, \quad (69)$$

we must have $\lambda > \frac{1}{n}$ for this to be the case. In contrast, if the matrix contains just one nonzero entry then the low-rank component should be set to zero. Since

$$\left\| \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \right\|_* = 1, \quad \left\| \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \right\|_1 = 1 \quad (70)$$

this will be the case if $\lambda < 1$. Finally, if the matrix just has a row of ones it is unclear whether we should assign it to the low-rank or the sparse component, so the value of the two terms in the cost function should have a similar value. Since

$$\left\| \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \right\|_* = \sqrt{n}, \quad \left\| \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \right\|_1 = n \quad (71)$$

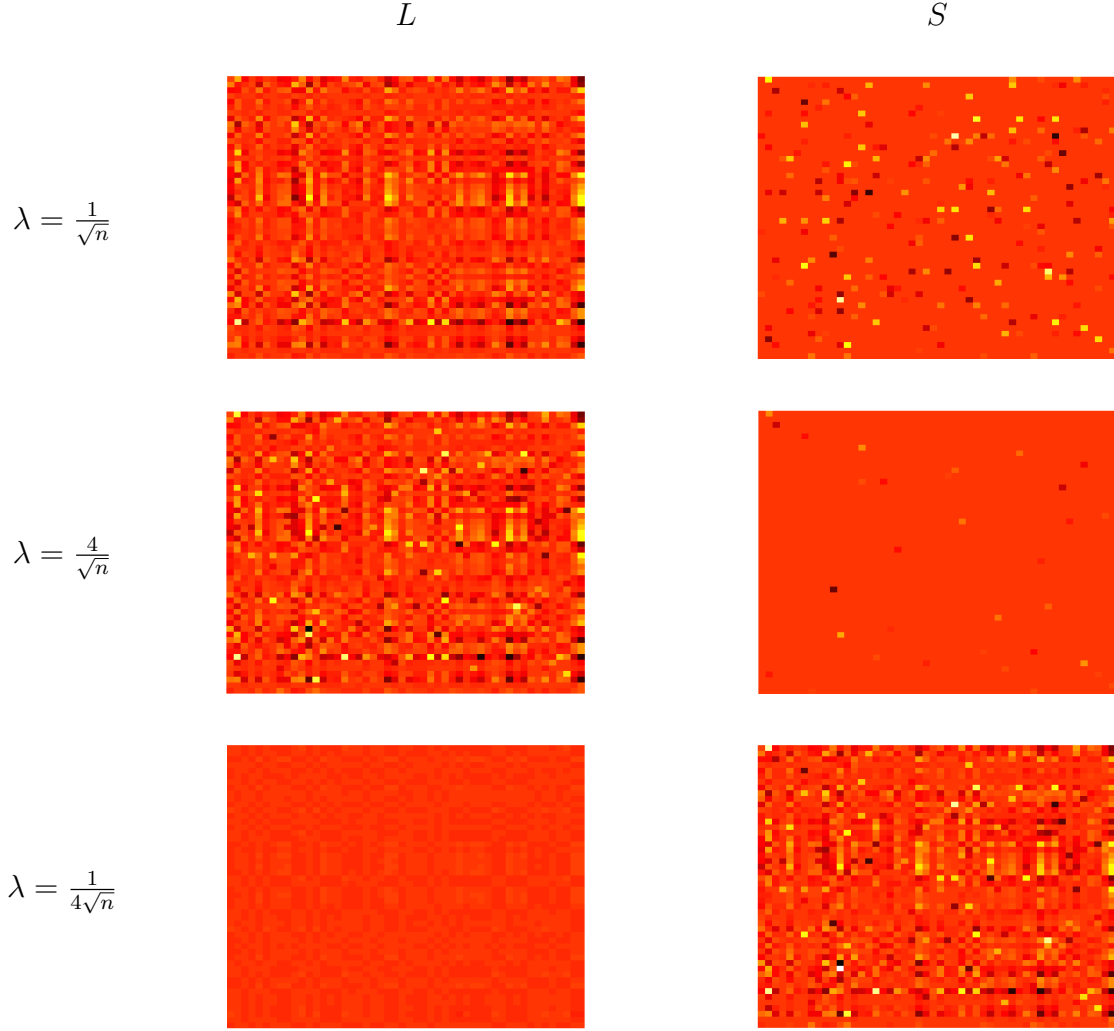


Figure 5: Results of solving Problem (68) for different values of the regularization parameter.

this suggests setting $\lambda \approx \frac{1}{\sqrt{n}}$.

Figure 5 shows the results of solving Problem (68) for different values of λ . If λ is too small, then it is *cheap* to increase the content of the sparse component, which won't be very sparse as a result. 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.

2.4 Theoretical guarantees

The following theorem from [2] provides theoretical guarantees for the decomposition of low-rank and sparse matrices using convex optimization. We omit some details in the statement of the result, which can be found in [2].

Theorem 2.1 (Exact decomposition via convex programming). *Let $M = L + S$ be a $n \times n$ matrix where L is rank r and has coherence μ and the support of S is distributed uniformly at random. Problem (68) with $\lambda = 1/\sqrt{n}$ recovers L and S exactly as long as the rank of L is of order n/μ up to logarithmic factors and the sparsity level of S is bounded by a certain constant times n^2 .*

As long as the low-rank component is incoherent and the support of the sparse component is uniformly distributed, convex programming allows to obtain a perfect decomposition up to values of the rank of L and the sparsity level of S that are essentially optimal.

The proof of this result is based on the construction of an approximate dual certificate. This certificate approximates the certificate described in the following proposition.

Proposition 2.2 (Dual certificate for robust PCA). *Let $M = L + S$, where $L = U\Sigma V^T$ is the singular-value decomposition of L and Ω is the support of S . A matrix Q of the form*

$$Q = UV^T + W = \lambda \text{sign}(S) + F \quad (72)$$

where

$$\|W\| < 1, \quad U^T W = 0, \quad W V = 0, \quad (73)$$

$$F_\Omega = 0, \quad \|F\|_\infty < \lambda, \quad (74)$$

is a dual certificate of the optimization problem

$$\min_{\tilde{L}, \tilde{S} \in \mathbb{R}^{m \times n}} \|\tilde{L}\|_* + \lambda \|\tilde{S}\|_1 \quad \text{such that } \tilde{L} + \tilde{S} = M. \quad (75)$$

Proof. Let us consider a feasible pair $L + L'$ and $S + S'$. Since $L + S = M$, $L + L' + S + S' = M$ implies $L' + S' = 0$. The conditions on Q imply that Q is a subgradient of the nuclear norm at L and that $\frac{1}{\lambda}Q$ is a subgradient of the ℓ_1 norm at S , which implies

$$\|L + L'\|_* + \lambda \|S + S'\|_1 \geq \|L\|_* + \langle Q, L' \rangle + \lambda \|S\|_1 + \lambda \left\langle \frac{1}{\lambda} Q, S' \right\rangle \quad (76)$$

$$= \|L\|_* + \lambda \|S\|_1 + \langle Q, L' + S' \rangle \quad (77)$$

$$= \|L\|_* + \lambda \|S\|_1. \quad (78)$$

Modifying the argument slightly to use the strict inequalities in (73) and (74) allows to prove that L and S are the unique solution. \square

2.5 Algorithms

In order to derive an algorithm to solve Problem (68), we will begin by considering a canonical problem with linear equality constraints

$$\text{minimize } f(x) \tag{79}$$

$$\text{subject to } Ax = y. \tag{80}$$

The Lagrangian of this problem is

$$\mathcal{L}(x, z) := f(x) + \langle z, Ax - y \rangle, \tag{81}$$

where z is a Lagrange multiplier or dual variable. The dual function is obtained by minimizing the Lagrangian over the primal variable x ,

$$g(z) := \inf_x f(x) + \langle z, Ax - y \rangle. \tag{82}$$

If strong duality holds, for a primal solution x^* and a dual solution z^* we have

$$f(x^*) = g(z^*) \tag{83}$$

$$= \inf_x \mathcal{L}(x, z^*) \tag{84}$$

$$\leq f(x^*). \tag{85}$$

This implies that if we know z^* , we can compute x^* by solving

$$\text{minimize } \mathcal{L}(x, z^*). \tag{86}$$

The dual-ascent method tries to find a solution to the dual problem by using gradient ascent on the dual function. In more detail, to compute the gradient of the dual function at z we first minimize the Lagrangian over the primal variable to obtain the minimizer \hat{x} . The gradient is then equal to

$$\nabla g(z) = A\hat{x} - y. \tag{87}$$

Algorithm 2.3 (Dual ascent). *We set an initial value $z^{(0)}$. Then we iterate between updating the primal and the dual variables.*

- *Primal-variable update*

$$x^{(k)} = \arg \min_x \mathcal{L}(x, z^{(k)}). \tag{88}$$

- *Dual-variable update*

$$z^{(k+1)} = z^{(k)} + \alpha^{(k)} (Ax^{(k)} - y) \tag{89}$$

for a step size $\alpha^{(k)} \geq 0$.

It turns out that the dual-ascent method is not very stable [1]. This may be tackled by defining an augmented Lagrangian

$$\mathcal{L}_\rho(x, z) := f(x) + \langle z, Ax - y \rangle + \frac{\rho}{2} \|Ax - y\|_2^2, \quad (90)$$

which corresponds to the Lagrangian of the modified (yet equivalent) problem

$$\text{minimize} \quad f(x) + \frac{\rho}{2} \|Ax - y\|_2^2 \quad (91)$$

$$\text{subject to} \quad Ax = y, \quad (92)$$

where $\rho > 0$ is a parameter. Applying dual ascent using the augmented Lagrangian yields the method of multipliers.

Algorithm 2.4 (Method of multipliers). *We set an initial value $z^{(0)}$. Then we iterate between updating the primal and the dual variables.*

- *Primal-variable update*

$$x^{(k)} = \arg \min_x \mathcal{L}_\rho(x, z^{(k)}). \quad (93)$$

- *Dual-variable update*

$$z^{(k+1)} = z^{(k)} + \rho (Ax^{(k)} - y). \quad (94)$$

Note that in the dual-variable update we have set the step size to ρ . This can be justified as follows. Setting $z^{(k+1)}$ so that

$$\nabla_x \mathcal{L}(x^{(k)}, z^{(k+1)}) = 0 \quad (95)$$

means that the Lagrangian of the original problem is minimized in each iteration. Given that

$$\nabla_x \mathcal{L}_\rho(x^{(k)}, z^{(k)}) = \nabla_x f(x^{(k)}) + A^T(z^{(k)} + \rho(Ax - y)) \quad (96)$$

and

$$\nabla_x \mathcal{L}_\rho(x^{(k)}, z^{(k)}) = 0, \quad (97)$$

setting $z^{(k+1)}$ as in (94) yields

$$\nabla_x \mathcal{L}(x^{(k)}, z^{(k+1)}) = \nabla_x f(x^{(k)}) + A^T(z^{(k)} + \rho(Ax - y)) \quad (98)$$

$$= 0. \quad (99)$$

Applying the same ideas to a composite objective function

$$\text{minimize} \quad f_1(x_1) + f_2(x_2) \quad (100)$$

$$\text{subject to} \quad Ax_1 + Bx_2 = y \quad (101)$$

yields the alternating direction method of multipliers (ADMM).

Algorithm 2.5 (Alternating direction method of multipliers). *We set an initial value $z^{(0)}$. Then we iterate between updating the primal and the dual variables.*

- *Primal-variable updates*

$$x_1^{(k)} = \arg \min_x \mathcal{L}_\rho \left(x, x_2^{(k-1)}, z^{(k)} \right), \quad (102)$$

$$x_2^{(k)} = \arg \min_x \mathcal{L}_\rho \left(x_1^{(k)}, x, z^{(k)} \right). \quad (103)$$

- *Dual-variable update*

$$z^{(k+1)} = z^{(k)} + \rho \left(Ax_1^{(k)} + Bx_2^{(k)} - y \right). \quad (104)$$

This description of ADMM is drawn from the review article [1], to which we refer the interested reader for more details. We now apply the method to Problem (68). In this case the augmented Lagrangian is of the form

$$\|L\|_* + \lambda \|S\|_1 + \langle Z, L + S - Y \rangle + \frac{\rho}{2} \|L + S - M\|_F^2. \quad (105)$$

The primal update for the low-rank component is obtained by applying Proposition 1.10.

$$L^{(k)} = \arg \min_L \mathcal{L}_\rho \left(L, S^{(k-1)}, Z^{(k)} \right) \quad (106)$$

$$= \arg \min_L \|L\|_* + \langle Z^{(k)}, L \rangle + \frac{\rho}{2} \|L + S^{(k-1)} - M\|_F^2 \quad (107)$$

$$= \mathcal{D}_{1/\rho} \left(\frac{1}{\rho} Z^{(k)} + S^{(k-1)} - M \right). \quad (108)$$

The primal update for the sparse component is obtained by recalling that softthresholding is the proximal operator for the ℓ_1 norm.

$$S^{(k)} = \arg \min_S \mathcal{L}_\rho \left(S, L^{(k-1)}, Z^{(k)} \right) \quad (109)$$

$$= \arg \min_S \lambda \|S\|_1 + \langle Z^{(k)}, S \rangle + \frac{\rho}{2} \|L^{(k-1)} + S - M\|_F^2 \quad (110)$$

$$= \mathcal{S}_{\lambda/\rho} \left(\frac{1}{\rho} Z^{(k)} + L^{(k-1)} - M \right). \quad (111)$$

This yields ADMM for robust PCA, also described under the name ALM (augmented Lagrangian method) in the literature [9].

Algorithm 2.6 (ADMM for robust PCA). *We set an initial value $Z^{(0)}$. Then we iterate between updating the primal and the dual variables.*

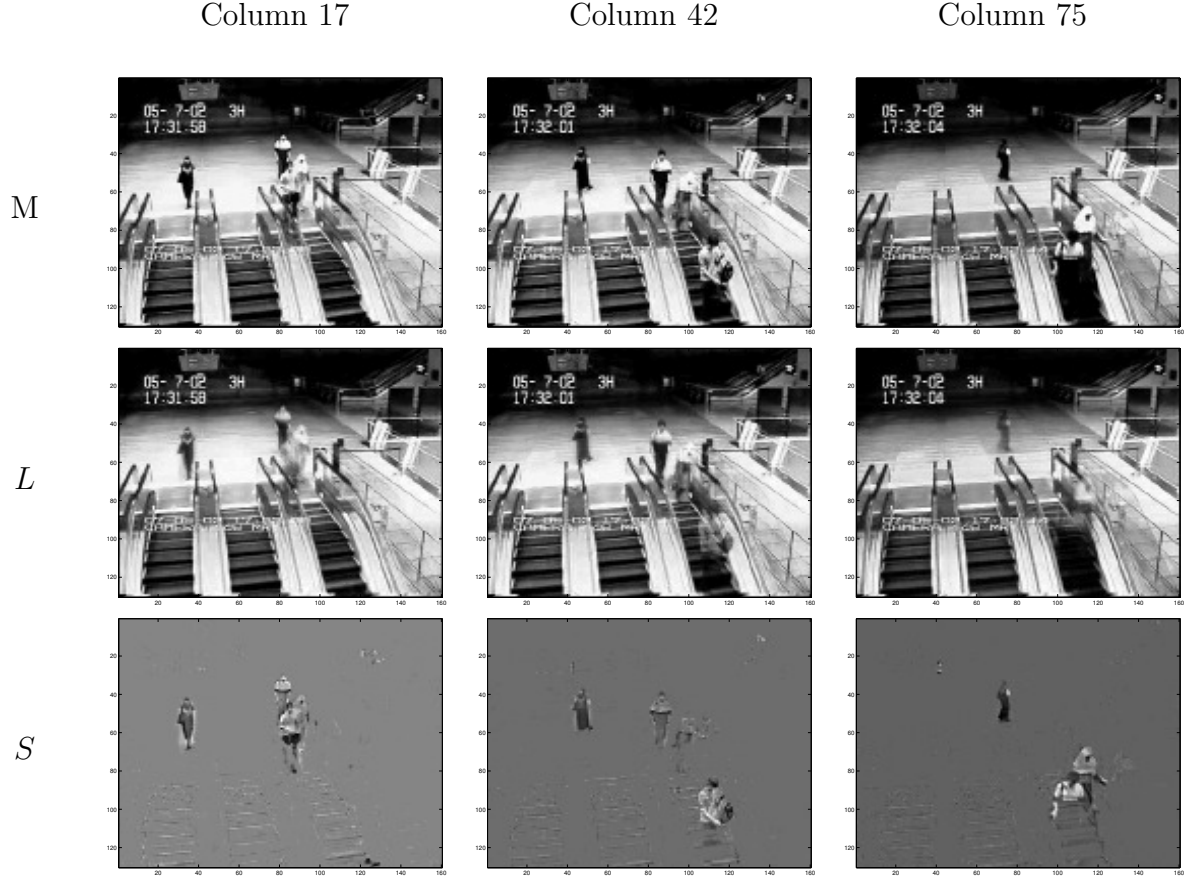


Figure 6: Background subtraction results from a video. This example is due to Stephen Becker. The code is available at <http://cvxr.com/tfocs/demos/rpca>.

- *Primal-variable updates*

$$L^{(k)} = \mathcal{D}_{1/\rho} \left(\frac{1}{\rho} Z^{(k)} + S^{(k-1)} - M \right), \quad (112)$$

$$S^{(k)} = \mathcal{S}_{\lambda/\rho} \left(\frac{1}{\rho} Z^{(k)} + L^{(k)} - M \right). \quad (113)$$

- *Dual-variable update*

$$Z^{(k+1)} = Z^{(k)} + \rho (L^{(k)} + S^{(k)} - M). \quad (114)$$

2.6 Background subtraction

In computer vision, the problem of background subtraction is that of separating the background and foreground of a video sequence. Imagine that we take a video of a static back-

ground. We then stack the video frames in a matrix M , where each column corresponds to a vectorized frame. If the background is completely static, then all the frames are equal to a certain vector $f \in \mathbb{R}^m$ (m is the number of pixels in each frame) and the matrix is rank 1

$$M = [f \ f \ \cdots \ f] = f [1 \ 1 \ \cdots \ 1]. \quad (115)$$

If the background is not completely static, but instead experiences gradual changes, then the matrix containing the frames will be approximately low rank. Now, assume that there are sudden events in the foreground. If these events occupy a small part of the field of view and do not last very long, then the corresponding matrix can be modeled as sparse (most entries are equal to zero). These observations motivate applying the robust PCA method to background subtraction. We stack the frames as columns of a matrix and separate the matrix into a low-rank and a sparse component. The results of applying this method to a real video sequence are shown in Figure 6.

References

Apart from the references cited in the text, the book [7] discusses matrix completion in Chapter 7.

- [1] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.
- [2] E. J. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):11, 2011.
- [3] E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717–772, 2009.
- [4] E. J. Candès and T. Tao. The power of convex relaxation: Near-optimal matrix completion. *Information Theory, IEEE Transactions on*, 56(5):2053–2080, 2010.
- [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] D. Gross. Recovering low-rank matrices from few coefficients in any basis. *Information Theory, IEEE Transactions on*, 57(3):1548–1566, 2011.
- [7] T. Hastie, R. Tibshirani, and M. Wainwright. *Statistical learning with sparsity: the lasso and generalizations*. CRC Press, 2015.
- [8] P. Jain, P. Netrapalli, and S. Sanghavi. Low-rank matrix completion using alternating minimization. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 665–674. ACM, 2013.

- [9] Z. Lin, M. Chen, and Y. Ma. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. *arXiv preprint arXiv:1009.5055*, 2010.

A Proofs

A.1 Proof of Proposition 1.5

The following simple lemma will be very useful. We omit the proof.

Lemma A.1. *For any $m \times n$ matrices A and B*

$$\text{trace}(AB) = \text{trace}(BA). \quad (116)$$

The proof relies on the following two lemmas.

Lemma A.2. *For any $Q \in \mathbb{R}^{m \times n}$, $U \in \mathbb{R}^{m \times m}$, $V \in \mathbb{R}^{n \times n}$, if $U^T U = I$ and $V^T V = I$ then*

$$\|UQV\| = \|Q\|. \quad (117)$$

Proof. By the definition of the operator norm,

$$\|UQV^T\| = \sup_{\|x\|_2=1} \|UQVx\|_2 \quad (118)$$

$$= \sup_{\|x\|_2=1} \sqrt{x^T V^T Q^T U^T U Q V x} \quad (119)$$

$$= \sup_{\|x\|_2=1} \sqrt{x^T V^T Q^T Q V x} \quad (120)$$

$$= \sup_{\|y\|_2=1} \sqrt{y^T Q^T Q y} \quad \text{because } \|x\|_2 = \|Vx\|_2 \quad (121)$$

$$= \|Q\|. \quad (122)$$

□

Lemma A.3. *For any $Q \in \mathbb{R}^{n \times n}$*

$$\max_{1 \leq i \leq n} |Q_{ii}| \leq \|Q\|. \quad (123)$$

Proof. We denote the standard basis vectors by e_i , $1 \leq i \leq n$. Since $\|e_i\|_2 = 1$,

$$\max_{1 \leq i \leq n} |Q_{ii}| \leq \max_{1 \leq i \leq n} \sqrt{\sum_{j=1}^n Q_{ji}^2} \quad (124)$$

$$= \max_{1 \leq i \leq n} \|Q e_i\|_2 \quad (125)$$

$$\leq \sup_{\|x\|_2=1} \|Q x\|_2. \quad (126)$$

□

By Lemma A.1, if the singular value decomposition of A is $U\Sigma V^T$ then

$$\sup_{\|B\| \leq 1} \text{tr}(A^T B) = \sup_{\|B\| \leq 1} \text{tr}(V \Sigma U^T B) \quad (127)$$

$$= \sup_{\|B\| \leq 1} \text{tr}(\Sigma B U^T V). \quad (128)$$

By Lemma A.2 $B U^T V$ has operator norm equal to $\|B\| = 1$. By Lemma A.3 this implies that its diagonal entries have magnitudes bounded by one. We conclude that

$$\sup_{\|B\| \leq 1} \text{tr}(A^T B) \leq \sup_{\{M \mid \max_{1 \leq i \leq n} |M_{ii}| \leq 1\}} \text{tr}(\Sigma M) \quad (129)$$

$$\leq \sup_{\{M \mid \max_{1 \leq i \leq n} |M_{ii}| \leq 1\}} \sum_{i=1}^n M_{ii} \sigma_i \quad (130)$$

$$\leq \sum_{i=1}^n \sigma_i \quad (131)$$

$$= \|A\|_*. \quad (132)$$

To complete the proof, we need to show that the equality holds. Note that UV^T has operator norm equal to one because its r singular values (recall that r is the rank of A) are equal to one. We have

$$\langle A, UV^T \rangle = \text{trace}(A^T UV^T) \quad (133)$$

$$= \text{trace}(V \Sigma U^T UV^T) \quad (134)$$

$$= \text{trace}(V^T V \Sigma) \quad \text{by Lemma A.1} \quad (135)$$

$$= \text{trace}(\Sigma) \quad (136)$$

$$= \|A\|_*. \quad (137)$$

A.2 Proof of Proposition 1.10

Due to the Frobenius norm term, the cost function is strictly convex. This implies that any point at which there exists a subgradient that is equal to zero is the solution to the optimization problem. The subgradients of the cost function at X are of the form,

$$X - Y + \tau G, \quad (138)$$

where G is a subgradient of the nuclear norm at X . If we can show that

$$\frac{1}{\tau} (Y - D_\tau(Y)) \quad (139)$$

is a subgradient of the nuclear norm at $D_\tau(Y)$ then $D_\tau(Y)$ is the solution.

Let us separate the singular-value decomposition of Y into the singular vectors corresponding to singular values greater than τ , denoted by U_0 and V_0 and the rest

$$Y = U \Sigma V^T \quad (140)$$

$$= [U_0 \ U_1] \begin{bmatrix} \Sigma_0 & 0 \\ 0 & \Sigma_1 \end{bmatrix} [V_0 \ V_1]^T. \quad (141)$$

Note that $D_\tau(Y) = U_0 (\Sigma_0 - \tau I) V_0^T$, so that

$$\frac{1}{\tau} (Y - D_\tau(Y)) = U_0 V_0^T + \frac{1}{\tau} U_1 \Sigma_1 V_1^T. \quad (142)$$

By construction all the singular values of $U_1 \Sigma_1 V_1^T$ are smaller than τ , so

$$\left\| \frac{1}{\tau} U_1 \Sigma_1 V_1^T \right\| \leq 1. \quad (143)$$

In addition, by definition of the singular-value decomposition $U_0^T U_1 = 0$ and $V_0^T V_1 = 0$. As a result, (142) is a subgradient of the nuclear norm at $D_\tau(Y)$ and the proof is complete.