Today, we move on to a new topic, namely that of nonconvex optimization in the context of low-rank matrix estimation.

9.1 Background

Given a subset $\mathcal{M} \subset \mathbb{R}^{d \times d}$ of all $d$-dimensional matrices, there are many matrix estimators that can be written in the form

$$\min_{M \in \mathcal{M}} \mathcal{L}_n(M) \quad \text{such that } \text{rank}(M) \leq r, \tag{9.1}$$

where the integer $r \in [1, d]$ is a target rank, and $\mathcal{L}_n$ is an empirical cost function defined by the observed data. In general, even when we assume that both $\mathcal{L}_n$ and $\mathcal{M}$ are convex, problems of this type are hard to solve due to the nonconvex rank constraint.

9.1.1 Some examples

Let us illustrate this set-up with some concrete examples.

Principal components analysis

A random vector $x_i \in \mathbb{R}^d$ is said to be drawn from a spiked covariance model if it can written in the form

$$x_i = F^* \sqrt{\Gamma} \xi_i + w_i$$

where $F^* \in \mathbb{R}^{d \times r}$ is a fixed matrix with orthonormal columns; $\Gamma = \text{diag}(\gamma_1, \ldots, \gamma_r)$ is a diagonal matrix with $\gamma_1 \geq \gamma_2 \geq \cdots \geq \gamma_r > 0$; $\xi_i \in \mathbb{R}^r$ is a zero-mean random vector with identity covariance, and $w_i$ is a zero-mean random vector, independent of $\xi_i$, and with identity covariance matrix $I_d$. Thus, the random vector $x_i \in \mathbb{R}^d$ is zero-mean with a covariance matrix of the form

$$\Sigma = (F^* \sqrt{\Gamma} \otimes F^* \sqrt{\Gamma}) + I_{d \times d}. \tag{9.2}$$

In the rank one case ($r = 1$), this model takes the simpler form

$$\Sigma = \gamma (f^* \otimes f^*) + I_{d \times d}, \tag{9.2}$$

1
where \( \gamma > 0 \) and \( f^* \in \mathbb{R}^d \). By construction, for any \( \gamma > 0 \), the vector \( f^* \) is the unique maximal eigenvector of \( \Sigma \) with eigenvalue \( \sigma_1(\Sigma) = \gamma + 1 \). All other eigenvalues of \( \Sigma \) are located at 1, so that we have an eigengap \( \sigma_1(\Sigma) - \sigma_2(\Sigma) = \gamma \).

A natural estimator of \( f^* \) can be constructed as follows. Let us form the sample covariance matrix \( \hat{\Sigma}_n = \frac{1}{n} \sum_{i=1}^{n} x_i x_i^T \), and then consider the optimization problem

\[
\min_{M \in \mathcal{M}} \mathcal{L}_n(M) = - \langle \langle \hat{\Sigma}_n, M \rangle \rangle := \text{trace}(\hat{\Sigma}_n^T M)
\]

such that \( \text{rank}(M) = 1 \),

where \( \mathcal{M} = \{ M \in \mathbb{S}^{d \times d} \mid \| M \|_{HS} \leq 1 \} \). Here the Hilbert-Schmidt norm is defined as

\[
\| M \|_{HS}^2 := \text{trace}(M^T M) = \sum_{i,j} \| M_{ij} \|^2.
\]

Because of the simple structure, this optimization problem has a closed form solution—namely \( \hat{M} = \hat{f} \otimes \hat{f} \) and \( \hat{f} \) is the maximal eigenvector of \( \hat{\Sigma}_n \). We can also deal with the case when \( r > 1 \) similarly.

Although the unstructured version of PCA leads to an easily solvable problem, this simplicity disappears once we have side constraints. Some examples include:

- **non-negative PCA**: the eigenvector \( f^* \) has nonnegative entries \( (f^*_i \geq 0) \), which implies the constraint \( M_{ij} \geq 0 \).
- **sparse PCA**: the eigenvector \( f^* \) is \( k \)-sparse, a constraint which can be relaxed (in similar fashion as with the Lasso) to the inequality \( \| f^* \|_1 \leq R \), which implies that \( \sum_{i,j=1}^d |M_{ij}| \leq R^2 \).

**Matrix completion**

Let \( \Omega \) denote the set of positions of the matrix \( M \) in which we receive observations

\[
\Omega = \{(a,b) \mid \text{make noisy observation of } M_{ab}\} \subset [d] \times [d]. \tag{9.3}
\]

A natural rank-constrained estimator is then given by

\[
\min_{M \in \mathcal{M}} \mathcal{L}_n(M) = \sum_{(a,b) \in \Omega} (Y_{ab} - M_{ab})^2 \quad \text{such that } \text{rank}(M) \leq r. \tag{9.4}
\]

If \( \Omega = [d] \times [d] \), then the objective function \( \mathcal{L}_n \) is equivalent to the HS norm of \( M \). Computing the SVD and truncating to the top \( r \) components then gives the optimal solution to the problem \( \text{[9.4]} \). In the more interesting case, the set \( \Omega \) is a strict subset of \( [d] \times [d] \), so that it is intractable to solve the problem \( \text{[9.4]} \) exactly.

**Matrix regression**

Matrix completion is a special case of the more general family of matrix regression problems. In the linear case, such problems are based on pairs \((y_i, X_i)\) that are linked via the observation model

\[
y_i = \langle X_i, M \rangle + w_i
\]
for $i = 1, \ldots, n$. Here each $X_i \in \mathbb{R}^{d \times d}$ is a matrix, and we observe a noisy version of the trace inner product of $X_i$ with $M$. Matrix completion is a special case of this set-up, in which each $X_i$ is a “mask matrix”—that is, a matrix with all zeroes except for a single one in the matrix entry that is observed for the $i^{th}$ sample.

### 9.1.2 Nuclear norm relaxation and its price

Since rank-constrained optimization problems of the form (9.1) are intractable in general, one is motivated to seek relaxations of it. A standard relaxation is to replace the rank constraint with a nuclear norm constraint. More precisely, we can write the SVD of $M$ as $M = U D V^T$ where $U$ and $V$ are $d$-dimensional orthonormal matrices of singular vectors, and $D$ is a diagonal matrix $D = \text{diag}(\sigma_1(M), \ldots, \sigma_d(M))$, with $\sigma_1(M) \geq \ldots \geq \sigma_d(M)$. With this notation, the nuclear norm is defined as $\|M\|_{\text{nuc}} = \sum_{i=1}^d |\sigma_i(M)|$. So instead of solving (9.4), we will find solution to

$$\min_{M \in \mathcal{M}} \mathcal{L}_n(M) \quad \text{such that } \|M\|_{\text{nuc}} \leq R,$$

where $R > 0$ is a user-chosen radius.

To understand why the nuclear norm encourages low-rank solutions, note that the rank of $M$ is encoded in the $\ell_0$-norm of the vector $\sigma(M) = (\sigma_1(M), \ldots, \sigma_d(M))$ of singular vectors—that is, $\text{rank}(M) = \|\sigma(M)\|_0$. The nuclear norm is simply the $\ell_1$-norm of this vector, so that it encourages sparsity, and hence low-rankness of the matrix.

### Price of nuclear norm relaxation

There is a large amount of literature on the statistical behavior of nuclear norm relaxations (e.g., [1, 5, 6, 4, 7, 3]). However, in terms of computing, although the resulting convex program (typically a semidefinite program) can be solved in polynomial time, it is often still computationally difficult. Part of the problem is that use of the nuclear norm is inherently a “lifting procedure”, one that increases the effective dimensionality of the problem. More concretely, if we optimize directly over the space of $r$-dimensional matrices, we have $\mathcal{O}(rd)$ free parameters. In contrast, the nuclear norm relaxation is formulated in a space with $\mathcal{O}(d^2)$ free parameters, which is much larger than $rd$ when the rank $r$ is small.

So can we seek a way to directly deal with the original non-convex problem? We now turn to a commonly used heuristic that respects the original dimensionality of the problem, and analyze it.

### 9.2 Direct factorized formulation

Suppose $M^*$ is rank $r$, then there exists $F^*$ and $G^* \in \mathbb{R}^{d \times r}$ such that $M^* = F^* \otimes G^* = F^*(G^*)^T$. We can then define the modified objective function

$$\tilde{\mathcal{L}}_n(F, G) = \mathcal{L}_n(F \otimes G).$$

This function is not jointly convex in general. As an illustration, let’s return to our earlier examples.
Matrix completion: In this case, we have
\[
\mathcal{L}_n(F \otimes G) = \sum_{(i,j) \in \Omega} (Y_{ij} - F \otimes G)^2.
\]
It is easy to check that this function is coordinate convex in \( F \) or \( G \) but not jointly convex. This fact suggests an iterative alternating direction minimization procedure to find \( M \), and these can be analyzed.

Principal component analysis: In this case, we know that \( M^* \) is both symmetric, positive semi-definite, and so has a decomposition of the form \( M^* = F^* \otimes F^* \). The associated cost function is given by
\[
\mathcal{L}_n(F \otimes G) = -\langle \hat{\Sigma}_n, F^* \otimes F^* \rangle.
\]
For instance, in the special case \( r = 1 \), we have \( F = f \in \mathbb{R}^d \), and
\[
\tilde{\mathcal{L}}_n(f) = f^T(-\hat{\Sigma}_n)f, \quad \|f\|_2 \leq 1.
\]
Notice that since \( \hat{\Sigma}_n \) is PSD, so \( \tilde{\mathcal{L}}_n \) is a concave function of \( f \). Concave functions are in general hard to minimize.

9.2.1 Projected gradient descent

In the remainder of the lecture, let us focus our attention on symmetric PSD matrices, and analyze problems of the form
\[
\min_{F \in \mathcal{F}} \tilde{\mathcal{L}}_n(F) := \mathcal{L}_n(F \otimes F). \quad (9.5)
\]
Here the constraint set \( \mathcal{F} \) is some convex subset of \( \mathbb{R}^{d \times r} \), chosen by the user. With reference to the original problem, it is typically chosen so that
\[
\mathcal{F} \times \mathcal{F} \approx \mathcal{M}.
\]
Assuming that \( \mathcal{L}_n \) is differentiable, the method of projected gradient descent generates a sequence of iterates via:
\[
F^{t+1} = \Pi_\mathcal{F}(F^t - s\nabla \tilde{\mathcal{L}}_n(F^t)) \quad (9.6)
\]
where \( \Pi_\mathcal{F} \) stands for projection onto \( \mathcal{F} \) under \( \| \cdot \|_{\text{HS}} \), and \( s > 0 \) is a stepsize parameter.

The question is when the PGD iterates \( \{F^t\} \) converge to a “good” solution. Here we will establish a result showing that they converge to an \( \epsilon_n \)-ball of the set
\[
\mathcal{E}(M^*) := \{ F^* \in \mathbb{R}^{d \times r} \mid F^* \otimes F^* = M^* \}. \quad (9.7)
\]
This is the best that we can expect since any matrix \( M \) typically has many factorizations. (In essence, the exact factorization itself is not identifiable.)
9.2.2 Restricted local descent and smoothness conditions

Let us first formulate some conditions on the cost function \( \tilde{L}_n \) under which the projected gradient descent updates are well-behaved. First, some useful notation: given any \( F \in \mathcal{F} \), let \( F_{\pi^*} = \Pi_{\mathcal{E}(M^*)}(F) \) to be the best approximation to \( F \) in \( \mathcal{E}(M^*) \) under HS norm.

\((\alpha, \epsilon_n)\)-local descent condition: The function \( \tilde{L}_n \) satisfies a \((\alpha, \epsilon_n)\)-local descent condition if

\[
\langle \langle \nabla \tilde{L}_n(F), F_{\pi^*} - F \rangle \rangle \leq -\alpha \|F - F_{\pi^*}\|_{HS}^2 + \alpha \epsilon_n^2 \quad \forall F \in B(\rho, F_{\pi^*}).
\] (9.8)

Note that \( \langle \langle \nabla \tilde{L}_n(F), F_{\pi^*} - F \rangle \rangle \) is the first order approximation of \( \tilde{L}_n(F_{\pi^*}) - \tilde{L}_n(F) \), so intuitively, inequality (9.8) means that the gradient descent method will make progress as long as \( F \) is not too close to \( F_{\pi^*} \).

Figure 9.2.2 provides an illustration of this condition. Our theorem will guarantee that if we initialize in a ball of radius \( \rho \) around some member of \( \mathcal{E}(M^*) \), then the PGD updates converge geometrically to an \( \epsilon \)-radius of \( \mathcal{E}(M^*) \).

One way of ensuring that the local descent condition holds is by imposing a strong convexity condition on \( \tilde{L}_n \) and requiring that the gradient \( \nabla \tilde{L}_n(F_{\pi^*}) \) is small. There are examples (such as PCA) where neither one of these conditions hold, but let’s explore the consequences in order to gain intuition.

**Lemma 9.2.1.** Suppose that \( \tilde{L}_n \) is \( c \)-strongly convex and \( \|\nabla \tilde{L}_n(F_{\pi^*})\|_{HS} \leq c \epsilon_n \). Then the \((\alpha, \epsilon_n)\)-local descent condition holds with \( \alpha = c/2 \).

**Proof.** By \( c \)-strong convexity, we have

\[
\langle \langle \nabla \tilde{L}_n(F) - \nabla \tilde{L}_n(F_{\pi^*}), F - F_{\pi^*} \rangle \rangle \geq c \|F - F_{\pi^*}\|_{HS}^2.
\]

Some algebra leads us to

\[
\langle \langle \nabla \tilde{L}_n(F), F - F_{\pi^*} \rangle \rangle \geq c \|F - F_{\pi^*}\|_{HS}^2 + \langle \langle \nabla \tilde{L}_n(F_{\pi^*}), F - F_{\pi^*} \rangle \rangle \geq c \|F - F_{\pi^*}\|_{HS}^2 - \|\nabla \tilde{L}_n(F_{\pi^*})\|_{HS} \|F - F_{\pi^*}\|_{HS} \] (9.9)
where the final step follows by Cauchy-Schwartz inequality.

Given our assumption that \( \| \nabla \tilde{L}_n(F_\pi^*) \|_{HS} \leq \alpha_n \), we then have

\[
\frac{\| \nabla \tilde{L}_n(F_\pi^*) \|_{HS}}{\sqrt{c}} (\sqrt{c} \| F - F_\pi^* \|_{HS}) \leq \frac{1}{2} (\frac{\| \nabla \tilde{L}_n(F_\pi^*) \|_{HS}}{\sqrt{c}})^2 + \frac{1}{2} (\sqrt{c} \| F - F_\pi^* \|_{HS})^2
\]

\[\leq \frac{c \epsilon_n^2}{2} + \frac{c}{2} \| F - F_\pi^* \|_{HS}^2.
\]

Plugging this expression into inequality (9.9) yields the claim.

The local descent condition is sufficient (on its own) to obtain convergence of the PGD algorithm. However, if we want to ensure fast convergence, then we also need a complementary form of restricted smoothness—closely related to a Lipschitz condition on the gradient \( \nabla \tilde{L}_n \).

**Local smoothness condition:** The function \( \tilde{L}_n \) satisfies a \((\alpha, \beta, \epsilon_n)\)-local smoothness condition if for any \( F, F' \in \mathcal{F}, F^* \in \mathcal{E}(M^*) \),

\[
| \langle \nabla \tilde{L}_n(F) - \nabla \tilde{L}_n(F'), F - F^* \rangle | \leq (\beta \| F - F' \|_{HS} + \alpha \epsilon_n) \| F - F^* \|_{HS}.
\]  

(9.10)

In order to gain intuition, note that this condition holds with \( \epsilon_n = 0 \) if \( \tilde{L}_n \) has a \( \beta \)-Lipschitz gradient—i.e., whenever \( \| \nabla \tilde{L}_n(F) - \nabla \tilde{L}_n(F') \|_{HS} \leq \beta \| F - F' \|_{HS} \). Again, allowing for \( \epsilon_n > 0 \) weakens the condition, since it no longer imposes a meaningful constraint when \( \| F - F' \|_{HS} \ll \epsilon_n \).

### 9.2.3 Convergence guarantee

For a rank \( r \) symmetric matrix \( M^* \in \mathcal{S}^{d \times d} \), given tolerance \( \tau \in (0, 1) \), suppose the following conditions are true:

- Condition \((\alpha, \beta, \epsilon_n)\) \{local descent, smoothness\} hold on \( \mathcal{B}(\rho, \mathcal{E}(M^*)) \).
- \( \rho = (1 - \tau) \sigma_r(M^*) \)
- sample size \( n \) is big enough so that \( \epsilon_n < \frac{1 - \tau}{4} \sigma_r(M^*) \).

Let’s now state our main theorem for today.

**Theorem 9.2.1.** If \( \tilde{L}_n \) satisfies the above three conditions, the PGD with constant step-size satisfies

\[
d^2_{HS}(F^t, \mathcal{E}(M^*)) \leq (1 - c \frac{\alpha^2}{\beta^2})^t d^2_{HS}(F^0, \mathcal{E}(M^*)) + 16 \epsilon_n^2.
\]

This theorem is essentially saying that the iterates go to some point in set \( \mathcal{E}(M^*) \) geometrically. Of course, if the error floor \( \epsilon_n \) is very large, the result is meaningless. In order for a result of this type to be useful, we need to prove that the descent/smoothness conditions hold with a “small” \( \epsilon_n \)—in the best case, it should be of the same order as the minimax rate of estimation! Given the length of the proof, we refer the reader to the paper [2] for full details. In the next lecture, we examine some specific examples.


