Chapter 3
Least Squares

3.1 ▪ "Solution" of Overdetermined Systems

Suppose that we are given a linear system of the form

$$\mathbf{A}\mathbf{x} = \mathbf{b},$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$ and $\mathbf{b} \in \mathbb{R}^m$. Assume that the system is *overdetermined*, meaning that $m > n$. In addition, we assume that $\mathbf{A}$ has a full column rank; that is, $\text{rank}(\mathbf{A}) = n$. In this setting, the system is usually *inconsistent* (has no solution) and a common approach for finding an approximate solution is to pick the solution resulting with the minimal squared norm of the residual $\mathbf{r} = \mathbf{A}\mathbf{x} - \mathbf{b}$:

$$\text{(LS)} \quad \min_{\mathbf{x} \in \mathbb{R}^n} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2.$$

Problem (LS) is a problem of minimizing a quadratic function over the entire space. The quadratic objective function is given by

$$f(\mathbf{x}) = \mathbf{x}^T \mathbf{A}^T \mathbf{A}\mathbf{x} - 2\mathbf{b}^T \mathbf{A}\mathbf{x} + \|\mathbf{b}\|^2.$$

Since $\mathbf{A}$ is of full column rank, it follows that for any $\mathbf{x} \in \mathbb{R}^n$ it holds that $\nabla^2 f(\mathbf{x}) = 2\mathbf{A}^T\mathbf{A} \succ 0$. (Otherwise, if $\mathbf{A}$ is not of full column, only positive semidefiniteness can be guaranteed.) Hence, by Lemma 2.41 the unique stationary point

$$\mathbf{x}_{\text{LS}} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{b} \tag{3.1}$$

is the optimal solution of problem (LS). The vector $\mathbf{x}_{\text{LS}}$ is called *the least squares solution* or *the least squares estimate* of the system $\mathbf{A}\mathbf{x} = \mathbf{b}$. It is quite common not to write the explicit expression for $\mathbf{x}_{\text{LS}}$ but instead to write the associated system of equations that defines it:

$$(\mathbf{A}^T\mathbf{A})\mathbf{x}_{\text{LS}} = \mathbf{A}^T\mathbf{b}.$$

The above system of equations is called *the normal system*. We can actually omit the assumption that the system is overdetermined and just keep the assumption that $\mathbf{A}$ is of full column rank. Under this assumption $m \geq n$, and in the case when $m = n$, the matrix $\mathbf{A}$ is nonsingular and the least squares solution is actually the solution of the linear system, that is, $\mathbf{A}^{-1}\mathbf{b}$.

Example 3.1. Consider the inconsistent linear system

$$\begin{aligned} x_1 + 2x_2 &= 0, \\ 2x_1 + x_2 &= 1, \\ 3x_1 + 2x_2 &= 1. \end{aligned}$$

We will denote by $\mathbf{A}$ and $\mathbf{b}$ the coefficients matrix and right-hand-side vector of the system. The least squares problem can be explicitly written as

$$\min_{x_1,x_2} (x_1 + 2x_2)^2 + (2x_1 + x_2 - 1)^2 + (3x_1 + 2x_2 - 1)^2.$$

Essentially, the solution to the above problem is the vector that yields the minimal sum of squares of the errors corresponding to the three equations. To find the least squares solution, we will solve the normal equations:

$$\begin{pmatrix} 1 & 2 \\ 2 & 1 \\ 3 & 2 \end{pmatrix}^T \begin{pmatrix} 1 & 2 \\ 2 & 1 \\ 3 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1 & 2 \\ 2 & 1 \\ 3 & 2 \end{pmatrix}^T \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix},$$

which are the same as

$$\begin{pmatrix} 14 & 10 \\ 10 & 9 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 5 \\ 3 \end{pmatrix}.$$

The solution of the above system is the least squares estimate:

$$\mathbf{x}_{\mathrm{LS}} = \begin{pmatrix} 15/26 \\ -8/26 \end{pmatrix}.$$

Note that

$$\mathbf{A}\mathbf{x}_{\mathrm{LS}} = \begin{pmatrix} -0.038 \\ 0.846 \\ 1.115 \end{pmatrix},$$

so that the residual vector containing the errors in each of the equations is

$$\mathbf{A}\mathbf{x}_{\mathrm{LS}} - \mathbf{b} = \begin{pmatrix} -0.038 \\ -0.154 \\ 0.115 \end{pmatrix}.$$

The total sum of squares of the errors, which is the optimal value of the least squares problem is $(-0.038)^2 + (-0.154)^2 + 0.115^2 = 0.038$. ∎

In MATLAB, finding the least squares solution is a very easy task.

MATLAB Implementation

To find the least squares solution of an overdetermined linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$ in MATLAB, the backslash operator \ should be used. Therefore, Example 3.1 can be solved by the following commands

```
>> A =[1,2;2,1;3,2];
>> b=[0;1;1];
>> format rational;
>> A\b
ans =
      15/26
      -4/13
```

3.2 ▪ Data Fitting

One area in which least squares is being frequently used is data fitting. We begin by describing the problem of linear fitting. Suppose that we are given a set of data points $(\mathbf{s}_i, t_i), i = 1, 2, \ldots, m$, where $\mathbf{s}_i \in \mathbb{R}^n$ and $t_i \in \mathbb{R}$, and assume that a linear relation of the form

$$t_i = \mathbf{s}_i^T \mathbf{x}, \quad i = 1, 2, \ldots, m,$$

approximately holds. In the least squares approach the objective is to find the parameters vector $\mathbf{x} \in \mathbb{R}^n$ that solves the problem

$$\min_{\mathbf{x}\in\mathbb{R}^n} \sum_{i=1}^{m} (\mathbf{s}_i^T \mathbf{x} - t_i)^2.$$

We can alternatively write the problem as

$$\min_{\mathbf{x}\in\mathbb{R}^n} \|\mathbf{S}\mathbf{x} - \mathbf{t}\|^2,$$

where

$$\mathbf{S} = \begin{pmatrix} -\, \mathbf{s}_1^T \,- \\ -\, \mathbf{s}_2^T \,- \\ \vdots \\ -\, \mathbf{s}_m^T \,- \end{pmatrix}, \qquad \mathbf{t} = \begin{pmatrix} t_1 \\ t_2 \\ \vdots \\ t_m \end{pmatrix}.$$

Example 3.2. Consider the 30 points in $\mathbb{R}^2$ described in the left image of Figure 3.1. The 30 x-coordinates are $x_i = (i-1)/29, i = 1, 2, \ldots, 30$, and the corresponding y-coordinates are defined by $y_i = 2x_i + 1 + \varepsilon_i$, where for every i, ε_i is randomly generated from a standard normal distribution with zero mean and standard deviation of 0.1. The MATLAB commands that generated the points and plotted them are

```
randn('seed',319);
d=linspace(0,1,30)';
e=2*d+1+0.1*randn(30,1);
plot(d,e,'*')
```

Note that we have used the command `randn('seed',sd)` in order to control the random number generator. In future versions of MATLAB, it is possible that this command will not be supported anymore, and will be replaced by the command `rng(sd,'v4')`. Given the 30 points, the objective is to find a line of the form $y = ax + b$ that best fits them. The corresponding linear system that needs to be "solved" is

$$\underbrace{\begin{pmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_{30} & 1 \end{pmatrix}}_{\mathbf{X}} \begin{pmatrix} a \\ b \end{pmatrix} = \underbrace{\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_{30} \end{pmatrix}}_{\mathbf{y}}.$$

The least squares solution of the above system is $(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$. In MATLAB, the parameters a and b can be extracted via the commands

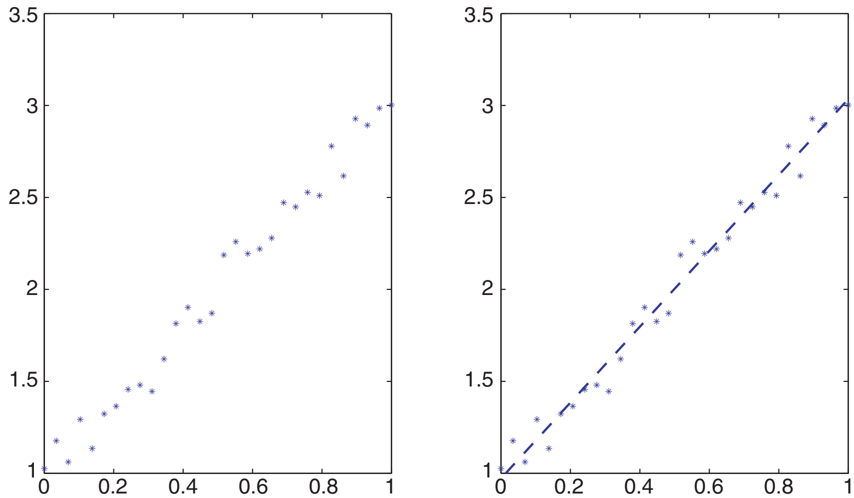

Figure 3.1. *Left image:* 30 *points in the plane. Right image: the points and the corresponding least squares line.*

```
>> u=[d,ones(30,1)]\e;
>> a=u(1),b=u(2)
a =
    2.0616
b =
    0.9725
```

Note that the obtained estimates of a and b are very close to the "true" a and b (2 and 1, respectively) that were used to generate the data. The least squares line as well as the 30 points is described in the right image of Figure 3.1. ∎

The least squares approach can be used also in nonlinear fitting. Suppose, for example, that we are given a set of points in $\mathbb{R}^2$: $(u_i, y_i), i = 1, 2, \ldots, m$, and that we know a priori that these points are approximately related via a polynomial of degree at most d; i.e., there exists $a_0, \ldots, a_d$ such that

$$\sum_{j=0}^{d} a_j u_i^j \approx y_i, \quad i = 1, \ldots, m.$$

The least squares approach to this problem seeks $a_0, a_1, \ldots, a_d$ that are the least squares solution to the linear system

$$\underbrace{\begin{pmatrix} 1 & u_1 & u_1^2 & \cdots & u_1^d \\ 1 & u_2 & u_2^2 & \cdots & u_2^d \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & u_m & u_m^2 & \cdots & u_m^d \end{pmatrix}}_{\mathbf{U}} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_d \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_m \end{pmatrix}.$$

The least squares solution is of course well-defined if the $m \times (d+1)$ matrix is of a full column rank. This of course suggests in particular that $m \geq d+1$. The matrix $\mathbf{U}$ consists of the first $d+1$ columns of the so-called *Vandermonde* matrix,

$$\begin{pmatrix} 1 & u_1 & u_1^2 & \cdots & u_1^{m-1} \\ 1 & u_2 & u_2^2 & \cdots & u_2^{m-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & u_m & u_m^2 & \cdots & u_m^{m-1} \end{pmatrix},$$

which is known to be invertible when all the u_is are different from each other. Thus, when $m \geq d+1$, and all the u_is are different from each other, the matrix $\mathbf{U}$ is of a full column rank.

3.3 ▪ Regularized Least Squares

There are several situations in which the least squares solution does not give rise to a good estimate of the "true" vector $\mathbf{x}$. For example, when $\mathbf{A}$ is underdetermined, that is, when there are fewer equations than variables, there are several optimal solutions to the least squares problem, and it is unclear which of these optimal solutions is the one that should be considered. In these cases, some type of prior information on $\mathbf{x}$ should be incorporated into the optimization model. One way to do this is to consider a penalized problem in which a *regularization function* $R(\cdot)$ is added to the objective function. The regularized least squares (RLS) problem has the form

$$\text{(RLS)} \quad \min_{\mathbf{x}} ||\mathbf{A}\mathbf{x}-\mathbf{b}||^2 + \lambda R(\mathbf{x}). \tag{3.2}$$

The positive constant λ is the *regularization parameter*. As λ gets larger, more weight is given to the regularization function.

In many cases, the regularization is taken to be quadratic. In particular, $R(\mathbf{x}) = ||\mathbf{D}\mathbf{x}||^2$ where $\mathbf{D} \in \mathbb{R}^{p\times n}$ is a given matrix. The quadratic regularization function aims to control the norm of $\mathbf{D}\mathbf{x}$ and is formulated as follows:

$$\min_{\mathbf{x}} ||\mathbf{A}\mathbf{x}-\mathbf{b}||^2 + \lambda||\mathbf{D}\mathbf{x}||^2.$$

To find the optimal solution of this problem, note that it can be equivalently written as

$$\min_{\mathbf{x}} \{f_{\mathrm{RLS}}(\mathbf{x}) \equiv \mathbf{x}^T(\mathbf{A}^T\mathbf{A} + \lambda\mathbf{D}^T\mathbf{D})\mathbf{x} - 2\mathbf{b}^T\mathbf{A}\mathbf{x} + ||\mathbf{b}||^2\}.$$

Since the Hessian of the objective function is $\nabla^2 f_{\mathrm{RLS}}(\mathbf{x}) = 2(\mathbf{A}^T\mathbf{A}+\lambda\mathbf{D}^T\mathbf{D}) \succeq \mathbf{0}$, it follows by Lemma 2.41 that any stationary point is a global minimum point. The stationary points are those satisfying $\nabla f_{\mathrm{RLS}}(\mathbf{x}) = \mathbf{0}$, that is,

$$(\mathbf{A}^T\mathbf{A} + \lambda\mathbf{D}^T\mathbf{D})\mathbf{x} = \mathbf{A}^T\mathbf{b}.$$

Therefore, if $\mathbf{A}^T\mathbf{A} + \lambda\mathbf{D}^T\mathbf{D} \succ \mathbf{0}$, then the RLS solution is given by

$$\mathbf{x}_{\mathrm{RLS}} = (\mathbf{A}^T\mathbf{A} + \lambda\mathbf{D}^T\mathbf{D})^{-1}\mathbf{A}^T\mathbf{b}. \tag{3.3}$$

Example 3.3. Let $\mathbf{A} \in \mathbb{R}^{3\times 3}$ be given by

$$\mathbf{A} = \begin{pmatrix} 2+10^{-3} & 3 & 4 \\ 3 & 5+10^{-3} & 7 \\ 4 & 7 & 10+10^{-3} \end{pmatrix}.$$

The matrix was constructed via the MATLAB code

```
B=[1,1,1;1,2,3];
A=B'*B+0.001*eye(3);
```

The "true" vector was chosen to be $\mathbf{x}_{\text{true}} = (1,2,3)^T$, and $\mathbf{b}$ is a noisy measurement of $\mathbf{A}\mathbf{x}_{\text{true}}$:

```
>> x_true=[1;2;3];
>> randn('seed',315);
>> b=A*x_true+0.01*randn(3,1)
b =
   20.0019
   34.0004
   48.0202
```

The matrix $\mathbf{A}$ is in fact of a full column rank since its eigenvalues are all positive (which can be checked, for example, by the MATLAB command `eig(A)`), and the least squares solution is given by $\mathbf{x}_{\text{LS}}$, whose value can be computed by

```
A\b
ans =
    4.5446
   -5.1295
    6.5742
```

$\mathbf{x}_{\text{LS}}$ is rather far from the true vector $\mathbf{x}_{\text{true}}$. One difference between the solutions is that the squared norm $\|\mathbf{x}_{\text{LS}}\|^2 = 90.1855$ is much larger then the correct squared norm $\|\mathbf{x}_{\text{true}}\|^2 = 14$. In order to control the norm of the solution we will add the quadratic regularization function $\|\mathbf{x}\|^2$. The regularized solution will thus have the form (see (3.3))

$$\mathbf{x}_{\text{RLS}} = (\mathbf{A}^T\mathbf{A} + \lambda\mathbf{I})^{-1}\mathbf{A}^T\mathbf{b}.$$

Picking the regularization parameter as $\lambda = 1$, the RLS solution becomes

```
>> x_rls=(A'*A+eye(3))\(A'*b)
x_rls =
    1.1763
    2.0318
    2.8872
```

which is a much better estimate for $\mathbf{x}_{\text{true}}$ than $\mathbf{x}_{\text{LS}}$. ∎

3.4 ▪ Denoising

One application area in which regularization is commonly used is *denoising*. Suppose that a noisy measurement of a signal $\mathbf{x} \in \mathbb{R}^n$ is given:

$$\mathbf{b} = \mathbf{x} + \mathbf{w}.$$

Here $\mathbf{x}$ is an unknown signal, $\mathbf{w}$ is an unknown noise vector, and $\mathbf{b}$ is the known measurements vector. The denoising problem is the following: Given $\mathbf{b}$, find a "good" estimate of $\mathbf{x}$. The least squares problem associated with the approximate equations $\mathbf{x} \approx \mathbf{b}$ is

$$\min \|\mathbf{x} - \mathbf{b}\|^2.$$

However, the optimal solution of this problem is obviously $\mathbf{x} = \mathbf{b}$, which is meaningless. This is a case in which the least squares solution is not informative even though the associated matrix—the identity matrix—is of a full column rank. To find a more relevant problem, we will add a regularization term. For that, we need to exploit some a priori information on the signal. For example, we might know in advance that the signal is smooth in some sense. In that case, it is very natural to add a quadratic penalty, which is the sum of the squares of the differences of consecutive components of the vector; that is, the regularization function is

$$R(\mathbf{x}) = \sum_{i=1}^{n-1} (x_i - x_{i+1})^2.$$

This quadratic function can also be written as $R(\mathbf{x}) = \|\mathbf{L}\mathbf{x}\|^2$, where $\mathbf{L} \in \mathbb{R}^{(n-1)\times n}$ is given by

$$\mathbf{L} = \begin{pmatrix} 1 & -1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & -1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & -1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 & -1 \end{pmatrix}.$$

The resulting regularized least squares problem is (with λ a given regularization parameter)

$$\min_{\mathbf{x}} \|\mathbf{x} - \mathbf{b}\|^2 + \lambda \|\mathbf{L}\mathbf{x}\|^2,$$

and its optimal solution is given by

$$\mathbf{x}_{\text{RLS}}(\lambda) = (\mathbf{I} + \lambda \mathbf{L}^T \mathbf{L})^{-1}\mathbf{b}. \tag{3.4}$$

Example 3.4. Consider the signal $\mathbf{x} \in \mathbb{R}^{300}$ constructed by the following MATLAB commands:

```
t=linspace(0,4,300)';
x=sin(t)+t.*(cos(t).^2);
```

Essentially, this is the signal given by $x_i = \sin(4\frac{i-1}{299}) + (4\frac{i-1}{299})\cos^2(4\frac{i-1}{299}), i = 1, 2, \ldots, 300$. A normally distributed noise with zero mean and standard deviation of 0.05 was added to each of the components:

```
randn('seed',314);
b=x+0.05*randn(300,1);
```

The true and noisy signals are given in Figure 3.2, which was constructed by the MATLAB commands.

```
subplot(1,2,1);
plot(1:300,x,'LineWidth',2);
subplot(1,2,2);
plot(1:300,b,'LineWidth',2);
```

In order to denoise the signal $\mathbf{b}$, we look at the optimal solution of the RLS problem given by (3.4) for four different values of the regularization parameter: $\lambda = 1, 10, 100, 1000$. The original true signal is denoted by a dotted line. As can be seen in Figure 3.3, as λ gets larger, the RLS solution becomes smoother. For $\lambda = 1$ the RLS solution $\mathbf{x}_{\text{RLS}}(1)$ is not

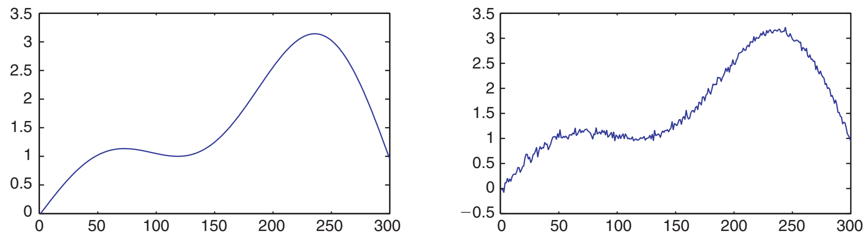

Figure 3.2. *A signal (left image) and its noisy version (right image).*

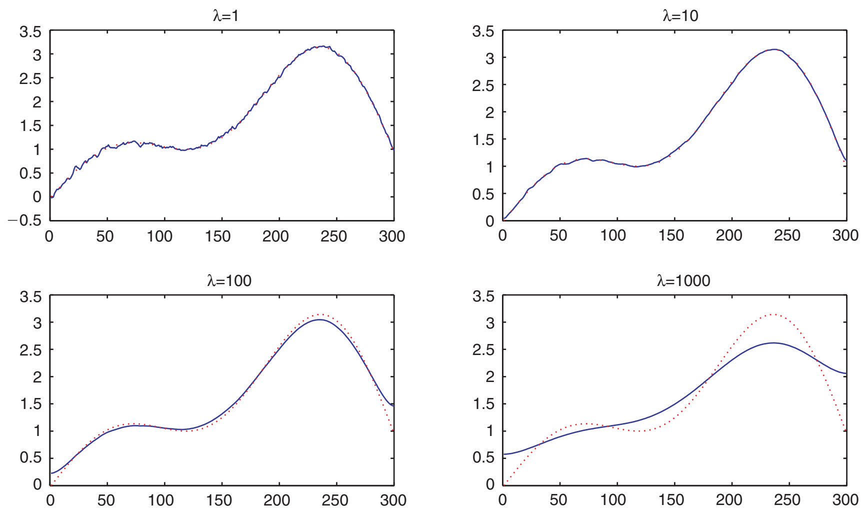

Figure 3.3. *Four reconstructions of a noisy signal by RLS solutions.*

smooth enough and is very close to the noisy signal **b**. For $\lambda = 10$ the RLS solution is a rather good estimate of the original vector **x**. For $\lambda = 100$ we get a smoother RLS signal, but evidently it is less accurate than $\mathbf{x}_{\text{RLS}}(10)$, especially near the boundaries. The RLS solution for $\lambda = 1000$ is very smooth, but it is a rather poor estimate of the original signal. In any case, it is evident that the parameter λ is chosen via a trade off between data fidelity (closeness of **x** to **b**) and smoothness (size of **Lx**). The four plots where produced by the MATLAB commands

```
L=zeros(299,300);
for i=1:299
    L(i,i)=1;
    L(i,i+1)=-1;
end

x_rls=(eye(300)+1*L'*L)\b;
x_rls=[x_rls,(eye(300)+10*L'*L)\b];
x_rls=[x_rls,(eye(300)+100*L'*L)\b];
x_rls=[x_rls,(eye(300)+1000*L'*L)\b];
figure(2)
```

```
for j=1:4
    subplot(2,2,j);
    plot(1:300,x_rls(:,j),'LineWidth',2);
    hold on
    plot(1:300,x,':r','LineWidth',2);
    hold off
    title(['\lambda=',num2str(10^(j-1))]);
end
```

∎

3.5 ▪ Nonlinear Least Squares

The least squares problem considered so far is also referred to as "linear least squares" since it is a method for finding a solution to a set of approximate linear equalities. There are of course situations in which we are given a system of *nonlinear* equations

$$f_i(\mathbf{x}) \approx c_i, \quad i = 1, 2, \ldots, m.$$

In this case, the appropriate problem is the *nonlinear least squares (NLS) problem*, which is formulated as

$$\min \sum_{i=1}^{m} (f_i(\mathbf{x}) - c_i)^2. \tag{3.5}$$

As opposed to linear least squares, there is no easy way to solve NLS problems. In Section 4.5 we will describe the Gauss–Newton method which is specifically devised to solve NLS problems of the form (3.5), but the method is not guaranteed to converge to the global optimal solution of (3.5) but rather to a stationary point.

3.6 ▪ Circle Fitting

Suppose that we are given m points $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_m \in \mathbb{R}^n$. The *circle fitting problem* seeks to find a circle

$$C(\mathbf{x}, r) = \{\mathbf{y} \in \mathbb{R}^n : \|\mathbf{y} - \mathbf{x}\| = r\}$$

that best fits the m points. Note that we use the term "circle," although this terminology is usually used in the plane ($n = 2$), and here we consider the general n-dimensional space $\mathbb{R}^n$. Additionally, note that $C(\mathbf{x}, r)$ is the boundary set of the corresponding ball $B(\mathbf{x}, r)$. An illustration of such a fit is given in Figure 3.4. The circle fitting problem has applications in many areas, such as archaeology, computer graphics, coordinate metrology, petroleum engineering, statistics, and more. The nonlinear (approximate) equations associated with the problem are

$$\|\mathbf{x} - \mathbf{a}_i\| \approx r, \quad i = 1, 2, \ldots, m.$$

Since we wish to deal with differentiable functions, and the norm function is not differentiable, we will consider the squared version of the latter:

$$\|\mathbf{x} - \mathbf{a}_i\|^2 \approx r^2, \quad i = 1, 2, \ldots, m.$$

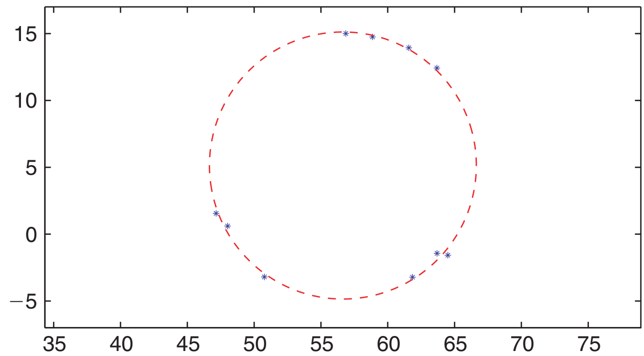

Figure 3.4. *The best circle fit (the optimal solution of problem (3.6)) of 10 points denoted by asterisks.*

The NLS problem associated with these equations is

$$\min_{\mathbf{x}\in\mathbb{R}^n, r\in\mathbb{R}_+} \sum_{i=1}^m (\|\mathbf{x}-\mathbf{a}_i\|^2 - r^2)^2. \tag{3.6}$$

From a first glance, problem (3.6) seems to be a standard NLS problem, but in this case we can show that it is in fact *equivalent* to a linear least squares problem, and therefore the global optimal solution can be easily obtained. We begin by noting that problem (3.6) is the same as

$$\min_{\mathbf{x},r} \left\{ \sum_{i=1}^m (-2\mathbf{a}_i^T\mathbf{x} + \|\mathbf{x}\|^2 - r^2 + \|\mathbf{a}_i\|^2)^2 : \mathbf{x}\in\mathbb{R}^n, r\in\mathbb{R} \right\}. \tag{3.7}$$

Making the change of variables $R = \|\mathbf{x}\|^2 - r^2$, the above problem reduces to

$$\min_{\mathbf{x}\in\mathbb{R}^n, R\in\mathbb{R}} \left\{ f(\mathbf{x},R) \equiv \sum_{i=1}^m (-2\mathbf{a}_i^T\mathbf{x} + R + \|\mathbf{a}_i\|^2)^2 : \|\mathbf{x}\|^2 \geq R \right\}. \tag{3.8}$$

Note that the change of variables imposed an additional relation between the variables that is given by the constraint $\|\mathbf{x}\|^2 \geq R$. We will show that in fact this constraint can be dropped; that is, problem (3.8) is equivalent to the linear least squares problem

$$\min_{\mathbf{x},R} \left\{ \sum_{i=1}^m (-2\mathbf{a}_i^T\mathbf{x} + R + \|\mathbf{a}_i\|^2)^2 : \mathbf{x}\in\mathbb{R}^n, R\in\mathbb{R} \right\}. \tag{3.9}$$

Indeed, any optimal solution $(\hat{\mathbf{x}}, \hat{R})$ of (3.9) automatically satisfies $\|\hat{\mathbf{x}}\|^2 \geq \hat{R}$ since otherwise, if $\|\hat{\mathbf{x}}\|^2 < \hat{R}$, we would have

$$-2\mathbf{a}_i^T\hat{\mathbf{x}} + \hat{R} + \|\mathbf{a}_i\|^2 > -2\mathbf{a}_i^T\hat{\mathbf{x}} + \|\hat{\mathbf{x}}\|^2 + \|\mathbf{a}_i\|^2 = \|\hat{\mathbf{x}} - \mathbf{a}_i\|^2 \geq 0, \qquad i = 1,\ldots,m.$$

Squaring both sides of the first inequality in the above equation and summing over i yield

$$f(\hat{\mathbf{x}},\hat{R}) = \sum_{i=1}^{m}\left(-2\mathbf{a}_i^T\hat{\mathbf{x}}+\hat{R}+\|\mathbf{a}_i\|^2\right)^2 > \sum_{i=1}^{m}\left(-2\mathbf{a}_i^T\hat{\mathbf{x}}+\|\hat{\mathbf{x}}\|^2+\|\mathbf{a}_i\|^2\right)^2 = f(\hat{\mathbf{x}},\|\hat{\mathbf{x}}\|^2),$$

showing that $(\hat{\mathbf{x}},\|\hat{\mathbf{x}}\|^2)$ gives a lower function value than $(\hat{\mathbf{x}},\hat{R})$, in contradiction to the optimality of $(\hat{\mathbf{x}},\hat{R})$. To conclude, problem (3.6) is equivalent to the least squares problem (3.9), which can also be written as

$$\min_{\mathbf{y}\in\mathbb{R}^{n+1}} \|\tilde{\mathbf{A}}\mathbf{y}-\mathbf{b}\|^2, \tag{3.10}$$

where $\mathbf{y} = \binom{\mathbf{x}}{R}$ and

$$\tilde{\mathbf{A}} = \begin{pmatrix} 2\mathbf{a}_1^T & -1 \\ 2\mathbf{a}_2^T & -1 \\ \vdots & \vdots \\ 2\mathbf{a}_m^T & -1 \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} \|\mathbf{a}_1\|^2 \\ \|\mathbf{a}_2\|^2 \\ \vdots \\ \|\mathbf{a}_m\|^2 \end{pmatrix}. \tag{3.11}$$

If $\tilde{\mathbf{A}}$ is of full column rank, then the unique solution of the linear least squares problem (3.10) is

$$\mathbf{y} = (\tilde{\mathbf{A}}^T\tilde{\mathbf{A}})^{-1}\tilde{\mathbf{A}}^T\mathbf{b}.$$

The optimal $\mathbf{x}$ is given by the first n components of $\mathbf{y}$ and the radius r is given by

$$r = \sqrt{\|\mathbf{x}\|^2 - R},$$

where R is the last (i.e., $(n+1)$th) component of $\mathbf{y}$. We summarize the above discussion in the following lemma.

Lemma 3.5. *Let $\mathbf{y} = (\tilde{\mathbf{A}}^T\tilde{\mathbf{A}})^{-1}\tilde{\mathbf{A}}^T\mathbf{b}$, where $\tilde{\mathbf{A}}$ and $\mathbf{b}$ are given in* (3.11)*. Then the optimal solution of problem* (3.6) *is given by $(\hat{\mathbf{x}},\hat{r})$, where $\hat{\mathbf{x}}$ consists of the first n components of $\mathbf{y}$ and $\hat{r} = \sqrt{\|\hat{\mathbf{x}}\|^2 - y_{n+1}}$.*

Exercises

3.1. Let $\mathbf{A}\in\mathbb{R}^{m\times n}, \mathbf{b}\in\mathbb{R}^n, \mathbf{L}\in\mathbb{R}^{p\times n}$, and $\lambda\in\mathbb{R}_{++}$. Consider the regularized least squares problem

$$\text{(RLS)} \qquad \min_{\mathbf{x}\in\mathbb{R}^n} \|\mathbf{A}\mathbf{x}-\mathbf{b}\|^2 + \lambda\|\mathbf{L}\mathbf{x}\|^2.$$

Show that (RLS) has a unique solution if and only if $\text{Null}(\mathbf{A})\cap\text{Null}(\mathbf{L}) = \{\mathbf{0}\}$, where here for a matrix $\mathbf{B}$, $\text{Null}(\mathbf{B})$ is the null space of $\mathbf{B}$ given by $\{\mathbf{x} : \mathbf{B}\mathbf{x} = \mathbf{0}\}$.

3.2. Generate thirty points $(x_i, y_i), i = 1,2,\ldots,30$, by the MATLAB code

```
randn('seed',314);
x=linspace(0,1,30)';
y=2*x.^2-3*x+1+0.05*randn(size(x));
```

Find the quadratic function $y = ax^2 + bx + c$ that best fits the points in the least squares sense. Indicate what are the parameters a, b, c found by the least squares

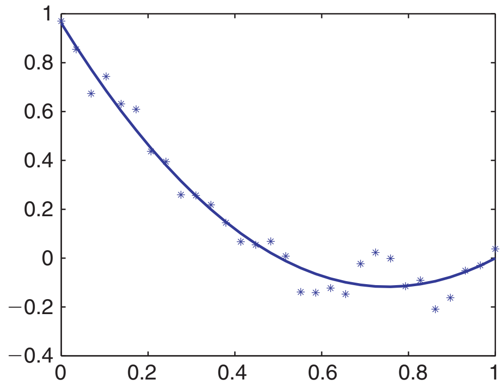

Figure 3.5. 30 *points and their best quadratic least squares fit.*

solution, and plot the points along with the derived quadratic function. The resulting plot should look like the one in Figure 3.5.

3.3. Write a MATLAB function `circle_fit` whose input is an $n \times m$ matrix $\mathbf{A}$; the columns of $\mathbf{A}$ are the m vectors in $\mathbb{R}^n$ to which a circle should be fitted. The call to the function will be of the form

```
[x,r]=circle_fit(A)
```

The output $(\mathbf{x}, r)$ is the optimal solution of (3.6). Use the code in order to find the best circle fit in the sense of (3.6) of the 5 points

$$\mathbf{a}_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad \mathbf{a}_2 = \begin{pmatrix} 0.5 \\ 0 \end{pmatrix}, \quad \mathbf{a}_3 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad \mathbf{a}_4 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad \mathbf{a}_5 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$