Chapter 2

Least Squares Rules

"Given a set of observations, which model parameters gives a model which approximates those up to the smallest sum of squared residuals?"

Least squares estimation serves as the blueprint and touchstone of most estimation techniques, and ideas should be mastered by any student. The purpose of this chapter is to survey results, to review the geometric insights and to elaborate on the numerical techniques used. In order to keep the exposition as simple as possible (but no simpler), we suppress for the moment the 'dynamic' nature of the studied systems, and focus on the static estimation problem.

2.1 Estimation by Optimizaztion

At first, let us introduce the main ideas behind different estimators. Informally:

We choose model parameters $\hat{\theta}$ which explain the data as well as possible.

To formalize this, we need to think about the following ideas:

(Model): What is a good set of functions $\{f_\theta\}$ which gives plausible models?

(Objective): In what sense do we need (formulate) the estimate be optimal?

(Optimization): How do we compute the minimization problem?

So the generic problem becomes

$$\hat{\theta} = \arg\min_{\theta} L_i(f_\theta(x_i, \theta), y_i). \tag{2.1}$$

This formulation makes the fundamental transition from observed input-output behavior to internal parameters which are not directly observed. Following the above discussion, this can be made explicit in various ways. Some common choices are

LS : The archetypical Least Squares (LS) estimation problem solves

$$\hat{\theta} = \arg\min_{\theta} \sum_{i=1}^{n} (f_\theta(x_i) - y_i)^2 \tag{2.2}$$

WLS : The Weighted Least Squares (WLS) estimation problem solves

$$\hat{\theta} = \arg\min_{\theta} \sum_{i=1}^{n} w_i \left(f_\theta(x_i) - y_i\right)^2 \tag{2.3}$$

TA : The Tchebychev Approximation (TA) problem solves

$$\hat{\theta} = \arg\min_{\theta} \max_{i=1,\ldots,n} |f_\theta(x_i) - y_i| \tag{2.4}$$

L1 : The L_1 estimation problem solves

$$\hat{\theta} = \arg\min_{\theta} \sum_{i=1}^{n} |f_\theta(x_i) - y_i| \tag{2.5}$$

L0 : A robust approximation problem solves

$$\hat{\theta} = \arg\min_{\theta} \sum_{i=1}^{n} |f_\theta(x_i) - y_i|_0 \tag{2.6}$$

where $|z|_0$ equals one if $z \neq 0$, and $|z|_0 = 0$ if and only if $z = 0$.

It depends on application specific considerations which one to choose. For example, if one wants to model in the context of peaks which are important to catch one may prefer (2.4). Fig. (2.1) exemplifies the different estimators.

2.2 Least Squares (OLS) estimates

2.2.1 Models which are Linear in the Parameters

This chapter studies a classical estimators for unknown parameters which occur linearly in a model structure. Such model structure will be referred to as Linear In the Parameters, abbreviated as LIP. At first we will give some examples in order to get an intuition about this class. Later sections then discuss how those parameters can be estimated using a least square argument. It is important to keep in the back of your mind that such least squares is not bound to LIP models, but then one ends up in general with less convenient (numerical, theoretical and conceptual) results.

Definition 2 (LIP) *A model for $\{y_i\}_i$ is linear in the unknowns $\{\theta_j\}_{j=1}^d$ if for each $y_i : i = 1, \ldots, n$, one has given values $\{x_{ij}\}_{j=1}^d$ such that*

$$y_i = \sum_{j=1}^{d} x_{ij}\theta_j + e_i, \tag{2.7}$$

and the terms $\{e_i\}$ are in some sense small. *Such model can be summarized schematically as*

$$\begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} x_{11} & \ldots & x_{1d} \\ \vdots & & \vdots \\ x_{n1} & \ldots & x_{nd} \end{bmatrix} \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_d \end{bmatrix} + \begin{bmatrix} e_1 \\ \vdots \\ e_n \end{bmatrix}, \tag{2.8}$$

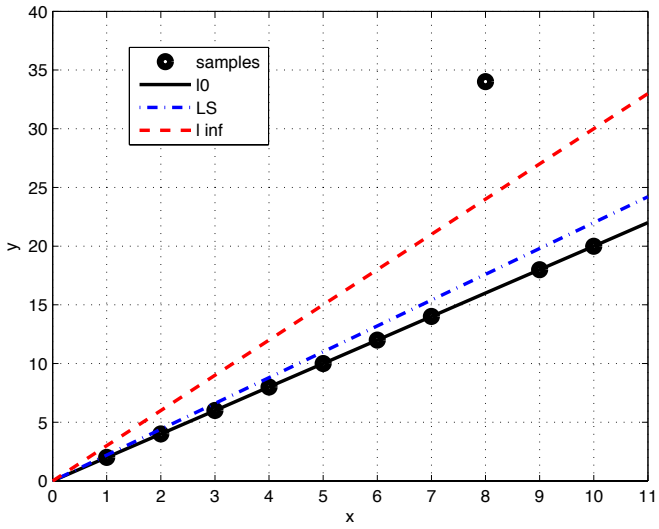


Figure 2.1: An example of an estimation problem. Assume that $\{f_\theta : \mathbb{R} \to \mathbb{R}\}$ equals the straight lines passing through the origin, with slope $\theta \in \mathbb{R}$. Assume that there are $n = 10$ samples (indicated by the black dots), with the 9th one containing a disturbance. Then the different estimators would give different solutions: (blue dashed-dotted line): the best θ according to the least squares criterion as in eq. (2.2), (red dashed line): the best θ according to the TA criterion as in eq. (2.4), (black solid line): the best θ according to the L_0 criterion as in eq. (2.6).

or in matrix notation

$$\mathbf{y} = \mathbf{X}\theta + \mathbf{e}. \tag{2.9}$$

where the matrix $\mathbf{X} \in \mathbb{R}^{n\times d}$*, the vectors* $\theta \in \mathbb{R}^d$ *and* $\mathbf{e} \in \mathbb{R}^n$ *are as in eq. (2.8). In case* d *is 'small' (compared to* n*), one refers to them as the* parameter *vector.*

If the input-output data we try to model can be captured in this form, the resulting problems , algorithms, analysis and interpretations become rather convenient. So the first step in any modeling task is to try to phrase the model formally in the LIP shape. Later chapters will study also problems who do not admit such parameterization. However, the line which models admit such parameterizations and which do not is not always intuitively clear. We support this claim with some important examples.

Example 1 (Constant Model) *Perhaps the simplest example of a linear model is*

$$y_t = \theta + e_t. \tag{2.10}$$

where $\theta \in \mathbb{R}$ *is the single parameter to estimate. This can be written as in eq. (2.7) as*

$$y_t = \mathbf{x}^T\theta + e_t. \tag{2.11}$$

where $\mathbf{x}_t = 1 \in \mathbb{R}$*, or in vector form as*

$$\mathbf{y} = 1_n\theta + \mathbf{e}, \tag{2.12}$$

where $\mathbf{y} = (y_1, \ldots, y_n)^T \in \mathbb{R}^n$*,* $\mathbf{e} = (e_1, \ldots, e_n)^T \in \mathbb{R}^n$ *and* $1_n = (1, \ldots, 1)^T \in \mathbb{R}^n$*. Hence, the 'inputs' take the constant value* 1*. This thinking is also used in order to express a model of* d *inputs* $\mathbf{X}_t = \mathbf{x}_t^T \in \mathbb{R}^d$ *for all* $t = 1, \ldots, n$ *with a constant intercept term* θ_0 *given as*

$$y_t = \mathbf{x}_t^T\theta + \theta_0 + e_t. \tag{2.13}$$

In matrix form one has

$$\mathbf{y} = \mathbf{X}'\theta' + \mathbf{e}, \tag{2.14}$$

where now $\theta' = (\theta^T\theta_0)^T \in \mathbb{R}^{d+1}$ *and* $\mathbf{X}' = \begin{bmatrix}\mathbf{X} & 1_n\end{bmatrix}$*.*

Example 2 (Polynomial Trend) *Assume the output has a polynomial trend of order smaller than* $m > 0$*, then it is good practice to consider the model*

$$y_t = \sum_{j=1}^{d} x_{tj}\theta_j + \sum_{k=0}^{m} t^k\theta'_k + e_t. = \mathbf{x}_t^T\theta + \mathbf{z}^T(t)\theta' + e_t, \tag{2.15}$$

where $\mathbf{z}(t) = (1, t, t^2, \ldots, t^m)^T \in \mathbb{R}^{m+1}$ *and* $\theta' = (\theta'_0, \theta'_1 \ldots, \theta_m)^T \in \mathbb{R}^{m+1}$*. Again, in matrix notation one has*

$$\mathbf{y} = \mathbf{X}'\theta' + \mathbf{e}. \tag{2.16}$$

where $\mathbf{X}'_t = (\mathbf{x}_t^T, 1, t, \ldots, t^m)$ *and* $\theta' = (\theta^T, \theta'_0, \theta'_1, \ldots, \theta'_m)^T \in \mathbb{R}^{d+m+1}$*.*

Example 3 (A Weighted sum of Exponentials) *It is crucial to understand that models which are linear in the parameters are not necessary linear in the covariates. For example, consider a nonlinear function*

$$y_t = f(\mathbf{x}_t) + e_t, \tag{2.17}$$

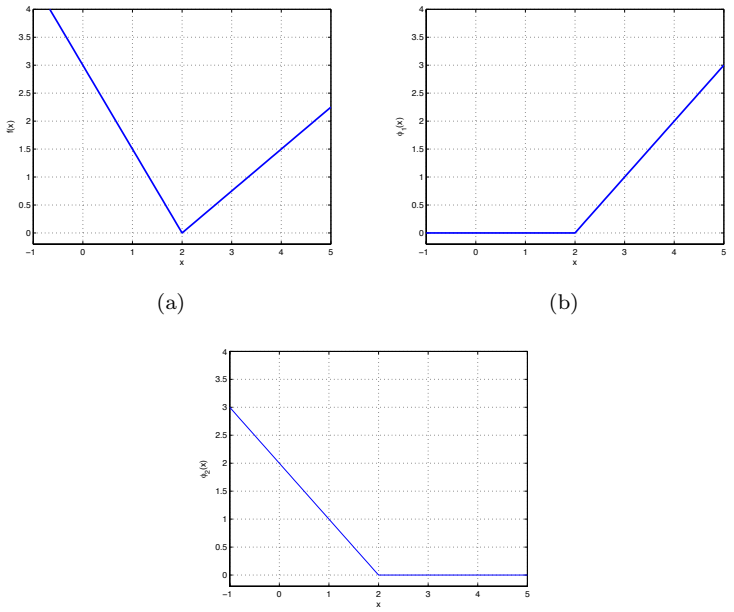
where $f : \mathbb{R}^d \to \mathbb{R}$ is any function. Then one can find an arbitrary good approximation to this model

$$y_t = \sum_{k=1}^{m} \varphi_k(\mathbf{x}_t)\theta_k + e_t = \varphi(\mathbf{x}_t)\theta + e_t, \tag{2.18}$$

where $\{\phi_1, \ldots, \phi_m\} \subset \{f : \mathbb{R}^d \to \mathbb{R}\}$ are an appropriate set of basis functions. *Then $\varphi(\mathbf{x}) = (\varphi_1(\mathbf{x}), \ldots, \varphi_m(\mathbf{x}))^T \in \mathbb{R}^m$. There are ample ways on which form of basis functions to consider. A method which often works is to work with exponential functions defined for any $k = 1, \ldots, m$ as*

$$\varphi_k(\mathbf{x}) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}_k\|}{\sigma_k}\right), \tag{2.19}$$

where $\sigma_k > 0$ and $\mathbf{x}_k \in \mathbb{R}^d$ is chosen suitably. Specific examples of basis functions are the orthogonal polynomials (e.g. Chebychev, Legendre or Laguerre polynomials). More involved sets lead to methods as wavelets, splines, orthogonal functions, or kernel methods.



(a) (b)

(c)

Figure 2.2: A Simple Example of a representation of a function $f(x)$ in panel (a) as the sum of two basis functions ϕ_1 (b) and ϕ_2 (c).

Example 4 (Dictionaries) *Elaborating on the previous example, it is often useful to model*

$$y_t = f(\mathbf{x}_t) + e_t, \tag{2.20}$$

as

$$y_t = \sum_{k=1}^{m} f_k(\mathbf{x}_t)\theta_k + e_t, \tag{2.21}$$

where the set $\{f_1, \ldots, f_m\}$ *is assumed to contain the unknown function* $f : \mathbb{R}^d \to \mathbb{R}$ *up to a scaling. If this is indeed the case, and the* $f \propto f_j$ *then eq. (2.20) can be represented as*

$$y_t = F_m(\mathbf{x}_t)\mathbf{e}_k a + e_t, \tag{2.22}$$

where $a \neq 0$ *is a constant,* $\mathbf{e}_k \in \{0,1\}^m$ *is the kth unit vector, that is* $\mathbf{e}_k = (0, \ldots, 1 \ldots, 0)^T$ *with unit on the kth position, and zero elsewhere.* F_m *denotes the dictionary, it is* $F_m(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_m(\mathbf{x}))^T \in \mathbb{R}^m$ *for all* $\mathbf{x} \in \mathbb{R}^d$.

In practice, if the model which is proposed to use for the identification experiment can be written as an expression which is linear in the parameters, then the subsequent analysis, inference and interpretation are often rather straightforward. The first challenge for successful identification is hence to phrase the problem in this form. It is however not always possible to phrase mathematical models in this form as indicated in the following example.

Example 5 (Nonlinear in the Parameters) *Consider the following model for observations* $\{y_t\}_{t=1}^n$

$$y_t = a\sin(bt + c), \tag{2.23}$$

where (a, b, c) *are unknown parameters. Then it is seen that the model is linear in* a*, but not in* b, c*. A way to circumvent this is to come up with plausible values* $\{b_1, \ldots, b_m\}$ *for* b*, and* $\{b_1, \ldots, b_m\}$ *for* c*, and to represent the model as*

$$y_t = \sum_{i,j=1}^{m} a_{i,j}\sin(b_i t + c_j), \tag{2.24}$$

where the model (2.23) is recovered when $a_{i,j} = a$ *when* $b_i = b$ *and* $c_j = c$*, and is zero otherwise.*

Example 6 (Quadratic in the Parameters) *Consider the following model for observations* $\{(y_t, x_t)\}_{t=1}^n$

$$y_t = ax_t + e_t + be_{t-1}, \tag{2.25}$$

where $\{e_t\}_t$ *are unobserved noise terms. Then we have cross-terms* $\{be_{t-1}\}$ *of unknown quantities, and the model falls not within the scope of models which are linear in the parameters.*

Other examples are often found in the context of grey-box models where theoretical study (often expressed in terms of PDEs) decide where to put the parameters. Nonlinear modeling also provide a fertile environment where models which are nonlinear in the parameters thrive. One could for example think of systems where nonlinear feedback occurs.

2.2.2 Ordinary Least Squares (OLS) estimates

Example 7 (Univariate LS) *Suppose we have given n samples $\{(x_i, y_i)\}_{i=1}^n$, where $x_i, y_i \in \mathbb{R}$. We suspect that they are strongly correlated in the sense that there is an unknown parameter $\theta \in \mathbb{R}$ such that $y_i \approx \theta x_i$ for all $i = 1, \ldots, n$. We hope to find a good approximation to θ by solving the following problem*

$$\hat{\theta} = \operatorname*{argmin}_{\theta \in \mathbb{R}} \sum_{i=1}^{n} (y_i - \theta x_i)^2. \tag{2.26}$$

At this stage, we have done the most work - i.e. converting the practical problem in a mathematical one - and what is left is mere technical. In this particular case, the solution is rather straightforward: first note that eq. (2.26) requires us to solve an optimization problem with (i) optimization criterion $J_n(\theta) = \sum_{i=1}^n (y_i - \theta x_i)^2$, and (ii) $\theta \in \mathbb{R}$ the variable to optimize over. It is easily checked that in general there is only a single optimum, and this one is characterized by the place where $\dfrac{\partial J_n(\theta)}{\partial \theta} = 0$. Working this one out gives

$$-2 \sum_{i=1}^{n} x_i y_i + 2 \sum_{i=1}^{n} x_i x_i \hat{\theta} = 0, \tag{2.27}$$

or

$$\hat{\theta} = \frac{\sum_{i=1}^n x_i y_i}{\sum_{i=1}^n x_i^2}, \tag{2.28}$$

That is, in case $\sum_{i=1}^n x_i^2 \neq 0$! This is a trivial remark in this case, but in general such conditions will play a paramount role in estimation problems.

Example 8 (Average) *Consider the simpler problem where we are after a variable θ which is 'close' to all datasamples $\{y_i\}_{i=1}^n$ taking values in $\mathbb{R}$. Again, we may formalize this as*

$$\hat{\theta} = \operatorname*{argmin}_{\theta \in \mathbb{R}} \sum_{i=1}^{n} (y_i - \theta)^2. \tag{2.29}$$

Do check that the solution is given as

$$\hat{\theta} = \frac{1}{n} \sum_{i=1}^{n} y_i. \tag{2.30}$$

In other words, the sample average is the optimal least squares approximation of a bunch of samples. This is no coincidence: we will explore the relation between sample averages, means and least squares optimal estimators in depth in later chapters. Note that in this particular case, there is no caveat to the solution, except for the trivial condition that $n > 0$.

The extension to more than one parameter is much easier by using matrix representations.

Example 9 (Bivariate Example) *Assume that we have a set of $n > 0$ couples $\{(x_{i,1}, x_{i,2}, y_i)\}_{i=1}^n$ to our disposition, where $x_{i,1}, x_{i,2}, y \in \mathbb{R}$. Assume that we try to 'fit a model'*

$$\theta_1 x_{i,1} + \theta_2 x_{i,2} + e_i = y_i, \tag{2.31}$$

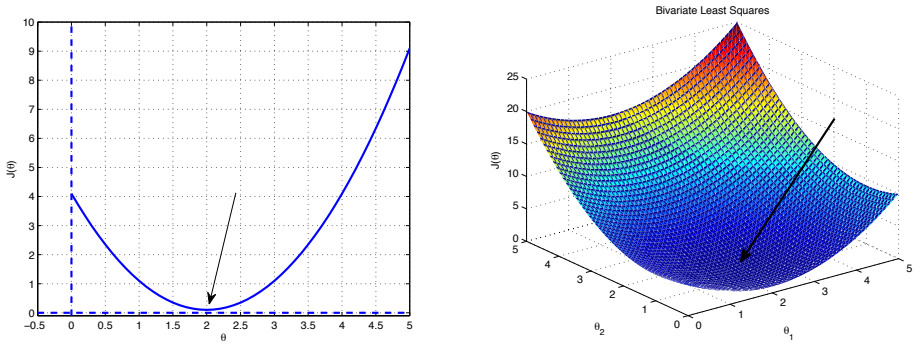


Figure 2.3: Illustration of the squared loss function in function of θ. The arrow indicates $\hat{\theta}$ where the minimum to $J(\theta)$ is achieved. Panel (a) shows the univariate case, or $\theta \in \mathbb{R}$ as in Example 2. Panel (b) shows the bivariate case, or $\theta \in \mathbb{R}^2$ as in Example 3.

where the unknown residuals $\{e_i\}_{i=1}^n$ *are thought to be 'small' in some sense. The Least Squares estimation problem is then written as*

$$(\hat{\theta}_1, \hat{\theta}_1) = \underset{\theta_1, \theta_2 \in \mathbb{R}}{\operatorname{argmin}} \sum_{i=1}^{n} (y_i - \theta_1 x_{i1} - \theta_2 x_{i2})^2. \tag{2.32}$$

This can be written out in matrix notation as follows. Let us introduce the matrix and vectors $\mathbf{X}_2 \in \mathbb{R}^{n \times 2}$, $\mathbf{y}, \mathbf{e} \in \mathbb{R}^n$ *and* $\theta \in \mathbb{R}^2$ *as*

$$\mathbf{X}_2 = \begin{bmatrix} x_{1,1} & x_{1,2} \\ x_{2,1} & x_{2,2} \\ \vdots & \vdots \\ x_{n,1} & x_{n,2} \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad \mathbf{e} = \begin{bmatrix} e_1 \\ y_2 \\ \vdots \\ e_n \end{bmatrix}, \quad \theta = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}. \tag{2.33}$$

Then the model (2.31) can be written as

$$\mathbf{X}_2 \theta + \mathbf{e} = \mathbf{y}. \tag{2.34}$$

and the Least Squares estimation problem (2.32) becomes

$$\hat{\theta} = \underset{\theta \in \mathbb{R}^2}{\operatorname{argmin}}\, J_n(\theta) = (\mathbf{X}_2 \theta - \mathbf{y})^T (\mathbf{X}_2 \theta - \mathbf{y}) \tag{2.35}$$

where the estimate $\hat{\theta} = (\hat{\theta}_1, \hat{\theta}_2)^T \in \mathbb{R}^2$ *is assumed to be unique. Taking the derivative of* $J_n(\theta)$ *and equating it to zero (why?) gives the following set of linear equations characterizing the solution:*

$$(\mathbf{X}_2^T \mathbf{X}_2)\theta = \mathbf{X}^T \mathbf{y}. \tag{2.36}$$

This set of linear equations has a unique solution in case the matrix $(\mathbf{X}_2^T \mathbf{X}_2)$ *is sufficiently 'informative'. In order to formalize this notion let us first consider some examples:*

1. *Assume* $x_{i,2} = 0$ *for all* $i = 1, \ldots, n$.

2. *Assume* $x_{i,2} = x_{i,1}$ *for all* $i = 1, \ldots, n$

3. *Assume* $x_{i,2} = ax_{i,1}$ *for all* $i = 1, \ldots, n$, *for a constant* $a \in \mathbb{R}$.

How does the matrix

$$(\mathbf{X}_2^T \mathbf{X}_2) = \begin{bmatrix} \sum_{i=1}^n x_{i,1}x_{i,1} & \sum_{i=1}^n x_{i,1}x_{i,2} \\ \sum_{i=1}^n x_{i,2}x_{i,1} & \sum_{i=1}^n x_{i,2}x_{i,2} \end{bmatrix}, \tag{2.37}$$

look like? Why does (2.36) give an infinite set of possible solutions in that case?

This reasoning brings us immediately to the more general case of $d \geq 1$ covariates. Consider the model which is *linear in the parameters*

$$y_i = \sum_{j=1}^{d} \theta_j x_{i,j} + e_i, \ \forall i = 1, \ldots, n. \tag{2.38}$$

Defining

$$\mathbf{X}_d = \begin{bmatrix} \mathbf{x}_1^T \\ \vdots \\ \mathbf{x}_n \end{bmatrix} = \begin{bmatrix} x_{1,1} & \ldots & x_{1,d} \\ \vdots & & \vdots \\ x_{n,1} & \ldots & x_{n,d} \end{bmatrix}, \tag{2.39}$$

or $\mathbf{X}_d \in \mathbb{R}^{n \times d}$ with $\mathbf{X}_{d,\ i,j} = x_{i,j}$. Note the orientation (i.e. the transposes) of the matrix as different texts use often a different convention. Equivalently, one may define

$$\mathbf{y} = \mathbf{X}_d \theta + \mathbf{e}, \tag{2.40}$$

where $\theta = (\theta_1, \ldots, \theta_d)^T \in \mathbb{R}^d$ and $\mathbf{y} = (y_1, \ldots, y_n)^T \in \mathbb{R}^n$ and $\mathbf{e} = (e_1, \ldots, e_n)^T \in \mathbb{R}^n$. The Least Squares estimation problem solves as before

$$\min_{\theta \in \mathbb{R}^d} J_n(\theta) = (\mathbf{X}_d \theta - \mathbf{y})^T (\mathbf{X}_d \theta - \mathbf{y}) \tag{2.41}$$

where the estimate is now $\hat{\theta} = (\hat{\theta}_1, \ldots, \hat{\theta}_d)^T \in \mathbb{R}^d$. Equating the derivative to zero gives a characterization of a solution θ in terms of a set of linear equations as

$$(\mathbf{X}_d^T \mathbf{X}_d)\theta = \mathbf{X}_d^T \mathbf{y}, \tag{2.42}$$

this set of equations is referred to as *the normal equations* associated to (2.41). Now it turns out that the condition for uniqueness of the solution to this set goes as follows.

Lemma 1 *Let* $n, d > 0$, *and given observations* $\{(x_{i,1}, \ldots, x_{i,d}, y_i)\}_{i=1}^n$ *satisfying the model (2.40) for a vector* $\mathbf{e} = (e_1, \ldots, e_n)^T \in \mathbb{R}^n$. *The solutions* $\{\theta\}$ *to the optimization problem (2.41) are characterized by the normal equations (2.42). This set contains a single solution if and only if (iff) there exists no* $\mathbf{w} \in \mathbb{R}^d$ *with* $\|\mathbf{w}\|_2 = 1$ *such that* $(\mathbf{X}_d^T \mathbf{X}_d)\mathbf{w} = 0_d$.

Proof: At first, assume there exists a $\mathbf{w} \in \mathbb{R}^d$ with $\|\mathbf{w}\|_2 = 1$ such that $(\mathbf{X}_d^T \mathbf{X}_d)\mathbf{w} = 0_d$. Then it is not too difficult to derive that there has to be many different solutions to (2.41). Specifically, let θ be a solution to the problem (2.41), then so is $\theta + a\mathbf{w}$ for any $a \in \mathbb{R}$.

Conversely, suppose there exists two different solutions, say θ and θ', then $\mathbf{w} \neq 0_d$ is such that $(\mathbf{X}_d^T \mathbf{X}_d)\mathbf{w} = 0_d$. This proofs the Lemma.

□

It is interesting to derive what the minimal value $J(\hat{\theta})$ will be when the optimum is achieved. This quantity will play an important role in later chapters on statistical interpretation of the result, and on model selection. Let's first consider a simple example:

Example 10 (Average, Ct'd) *Consider the again the case where we are after a variable θ which is 'close' to all datasamples $\{y_i\}_{i=1}^n$ taking values in $\mathbb{R}$, or*

$$\hat{\theta} = \underset{\theta \in \mathbb{R}}{\operatorname{argmin}} J(\theta) = \sum_{i=1}^{n} (y_i - \theta)^2. \tag{2.43}$$

The solution is characterized as $\hat{\theta} = \frac{1}{n}\sum_{i=1}^n y_i$. Then the achieved minimal value $J(\hat{\theta})$ equals

$$J(\hat{\theta}) = \sum_{i=1}^{n} (y_i - \hat{\theta})^2 = \sum_{i=1}^{n} y_i^2 - 2\sum_{i=1}^{n} y_i\hat{\theta} + \hat{\theta}^2 = \sum_{i=1}^{n} y_i^2 - \left(\sum_{i=1}^{n} y_i\right)^2. \tag{2.44}$$

as verified by straightforward calculation.

In general, the value $J(\hat{\theta})$ is expressed as follows.

2.2.3 Ridge Regression

What to do in case multiple solutions exists? It turns out that there exists two essentially different approaches which become almost as elementary as the OLS estimator itself. we consider again the estimators solving

$$\Omega = \underset{\theta \in \mathbb{R}^d}{\operatorname{argmin}} J_n(\theta) = (\mathbf{X}_d\theta - \mathbf{y})^T(\mathbf{X}_d\theta - \mathbf{y}), \tag{2.45}$$

where $\Omega \subset \mathbb{R}^d$ is now a set.

- Select the smallest solution amongst the set of all solutions Ω.
- Modify the objective such that there is always a unique solution.

The first approach is very much a procedural approach, and details will be given in the section about numerical tools. It is noteworthy that such approach is implemented through the use of the pseudo-inverse.

The second approach follows a more general path. In its simplest form the modified optimization problem becomes

$$\min_{\theta \in \mathbb{R}^d} J_n^\gamma(\theta) = (\mathbf{X}_d\theta - \mathbf{y})^T(\mathbf{X}_d\theta - \mathbf{y}) + \gamma\theta^T\theta, \tag{2.46}$$

where $\gamma \geq 0$ regulates the choice of how the terms (i) $\|\mathbf{X}_d\theta - \mathbf{y}\|_2^2$ and (ii) $\|\theta\|_2^2$ are traded off. If γ is chosen large, one emphasizes 'small' solutions, while the corresponding first term (i) might be suboptimal. In case $\gamma \approx 0$ one enforces the first term to be minimal, while imposing a preference on all vectors $\{\theta\}$ minimizing this term. It is easy to see that in case $\gamma > 0$ there is only a single solution to (2.46). Indeed equating the derivative of (2.46) to zero would give the following characterization of a solution $\theta \in \mathbb{R}^d$

$$(\mathbf{X}_d^T\mathbf{X}_d + \gamma I_d)\theta = \mathbf{X}_d^T\mathbf{y}, \tag{2.47}$$

and it becomes clear that no $\mathbf{w} \in \mathbb{R}^d$ with $\|\mathbf{w}\|_2 = 1$ exist such that $(\mathbf{X}_d^T\mathbf{X}_d + \gamma I_d)\mathbf{w} = 0_d$. in case there is only a single θ which achieves the minimum to $\|\mathbf{X}_d\theta - \mathbf{y}\|_2^2$, a nonzero γ would give a slightly different solution to (2.46), as opposed to this θ. It would be up to the user to control this difference, while still ensuring uniqueness of the solution when desired.

Recently, a related approach came into attention. Rather than adding a small jitter term $\theta^T\theta$, it is often advantageous to use a jitter term $\|\theta\|_1 = \sum_{j=1}^d |\theta_j|$. The objective then becomes

$$\min_{\theta\in\mathbb{R}^d} J_n^\gamma(\theta) = (\mathbf{X}_d\theta - \mathbf{y})^T(\mathbf{X}_d\theta - \mathbf{y}) + \gamma\|\theta\|_1. \tag{2.48}$$

The solution to this problem can be computed efficiently using tools of numerical optimization as surveyed in Chapter 15. Why to prefer (2.48) over (2.46)? Denote the estimates resulting from solving (2.46) as $\hat{\theta}_2$, and the estimates based on the same $\mathbf{X}, \mathbf{y}$ obtained by solving (2.48) as $\hat{\theta}_1$. Then the main insight is that the latter will often contain zero values in the vector $\hat{\theta}_1$. Those indicate often useful information on the problem at hand. For example, they could be used for selecting relevant inputs, orders or delays. Solution $\hat{\theta}_2$ in contrast will rarely contain zero parameters. But then, it is numerically easier to solve (2.46) and to characterize theoretically the optimum.

2.3 Numerical Tools

The above techniques have become indispensable tools for researchers involved with processing data. Their solutions are characterized in terms of certain matrix relations. The actual power of such is given by the available tools which can be used to solve this problems numerically in an efficient and robust way. This section gives a brief overview of how this works.

2.3.1 Solving Sets of Linear Equations

A central problem is how a set of linear equations can be solved. That is, given coefficients $\{a_{ij}\}_{i=1,\ldots,d,j=1,\ldots,d'}$ and $\{b_i\}_{i=1}^d$, find scalars $\{\theta_i \in \mathbb{R}\}_{i=1}^{d'}$ such that

$$\begin{cases} a_{11}\theta_1 + \ldots a_{1d'}\theta_{d'} = b_1 \\ \vdots \\ a_{d1}\theta_1 + \ldots a_{dd'}\theta_1 = b_d. \end{cases} \tag{2.49}$$

This set of linear equations can be represented in terms of matrices as $\mathbf{b} = (b_1, \ldots, b_d)^T \in \mathbb{R}^d$ and

$$\mathbf{A} = \begin{bmatrix} a_{11} & \ldots & a_{1d'} \\ \vdots & & \vdots \\ a_{d1} & \ldots & a_{dd'} \end{bmatrix}. \tag{2.50}$$

The set of linear equations is then written shortly as

$$\mathbf{A}\theta = \mathbf{b}. \tag{2.51}$$

Now we discriminate between 3 different cases:

$d < d'$ Then the matrix $\mathbf{A}$ looks *fat*, and the system is underdetermined. That is, there are an infinite set of possible solutions: there are not enough equality conditions in order to favor a single solution.

$d > d'$ Then the matrix $\mathbf{A}$ looks *tall*, and the system is in general overdetermined. That is, there is in general no solution vector $\theta = (\theta_1, \ldots, \theta_{d'})^T \in \mathbb{R}^{d'}$ which satisfies all equations simultaneously. Note that in certain (restrictive) conditions on the equality constraints, it is possible for a solution to exist.

$d = d'$ This implies that $\mathbf{A} \in \mathbb{R}^{d \times d}$ is square. In general, there is exactly one vector $\theta = (\theta_1, \ldots, \theta_d) \in \mathbb{R}^d$ which obeys all the equality constraints at once. In some cases this solution is however not unique.

As explained in the previous section, a vector $\theta \in \mathbb{R}^{d'}$ can satisfy more than d' equalities (i.e. $d > d'$) only when at least one of the equalities can be written as a linear combination of the other equalities. Numerical solutions to solve this system of equations include the Gauss Elimination or the Gauss-Newton algorithms. It is found that both theoretical as well as practical advantages are achieved when using a Conjugate Gradient Algorithm (CGA). Plenty of details of such schemes can be found in standard textbooks on numerical analysis and optimization algorithms, see e.g. [].

In case that there is no exact solution to the set of equality constraints, one can settle for the next best thing: the best approximate solution. If 'best' is formalized in terms of least squares norm of the errors needed to make the equalities hold approximatively, one gets

$$\min_{\theta} \sum_{i=1}^{d} \left(\sum_{j=1}^{d} a_{ij}\theta_j - b_i \right)^2 = \min_{\theta} \|\mathbf{A}\theta - \mathbf{b}\|_2^2. \tag{2.52}$$

which can again be solved as ... an OLS problem, where the solution in turn is given by solving according normal equations $\mathbf{A}^T\mathbf{A}\theta = \mathbf{A}^T\mathbf{b}$ of size d'.

A crucial property of a matrix is its rank, defined as follows.

Definition 3 (Rank) *A matrix $\mathbf{A} \in \mathbb{C}^{n \times d}$ with $n \geq d$ is rank-deficient if there exists a nonzero vector $\mathbf{x} \in \mathbb{C}^d$ such that*

$$\mathbf{A}\mathbf{x} = 0_n \tag{2.53}$$

where $0_n \in \mathbb{R}^n$ denotes the all-zero vector. Then the rank of a matrix is defined as the number of nonzero linear independent vectors $\{\mathbf{x}_1, \ldots, \mathbf{x}_r\} \subset \mathbb{R}^d$ which have that $\mathbf{A}\mathbf{x}_i \neq 0_n$, or

$$\text{rank}(\mathbf{A}) = \max \left| \{\mathbf{x}_i \in \mathbb{R}^d \quad s.t. \quad \mathbf{A}\mathbf{x}_i \neq 0_n, \mathbf{x}_i^T\mathbf{x} = \delta_{i-j}, \ \forall i, j = 1, \ldots, r\} \right| \leq \min(n, d). \tag{2.54}$$

2.3.2 Eigenvalue Decompositions

The previous approaches are mostly procedural, i.e. when implementing them the solution is computed under suitable conditions. However, a more fundamental approach is based on characterizing the properties of a matrix. In order to achieve this, the notion of a n Eigen Value Decomposition (EVD) is needed.

Definition 4 (Eigenpair) *Given a matrix $\mathbf{A} \in \mathbb{C}^{d' \times d}$ which can contain complex values. Then a vector $\mathbf{x} \in \mathbb{R}^d$ with $\|\mathbf{x}\|_2 = 1$ and corresponding value $\lambda \in \mathbb{C}$ constitutes an eigenpair $(\mathbf{x}, \lambda) \in \mathbb{C}^d \times \mathbb{C}$ if they satisfy*

$$\mathbf{A}\mathbf{x} = \lambda\mathbf{x}, \tag{2.55}$$

That is, if the matrix $\mathbf{A}$ *applied* to the vector $\mathbf{x}$ transforms into a rescaled version the same vector. It is intuitively clear that working with such eigenpairs simplifies an analysis since it reduces to working with scalars instead. Suppose we have d' such eigenpairs $\{(\mathbf{x}_i, \lambda_i)\}_{i=1}^n$, then those can be represented in matrix formulation as

$$\mathbf{A}\left[\mathbf{x}_1, \ldots, \mathbf{x}_{d'}\right] = \left[\lambda_1\mathbf{x}_1, \ldots, \lambda_{d'}\mathbf{x}_{d'}\right] = \left[\mathbf{x}_1, \ldots, \mathbf{x}_{d'}\right] \operatorname{diag}(\lambda_1, \ldots, \lambda_{d'}), \tag{2.56}$$

or

$$\mathbf{A}\mathbf{X} = \mathbf{X}\Lambda. \tag{2.57}$$

where $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_{d'}) \in \mathbb{C}^{d' \times d'}$ is a diagonal matrix, and $\mathbf{X} = \left[\mathbf{x}_1, \ldots, \mathbf{x}_{d'}\right] = \left[\mathbf{x}_1, \ldots, \mathbf{x}_{d'}\right] \in \mathbb{R}^{d \times d'}$.

The eigenvalues have a special form when the matrix $\mathbf{A}$ has special structure. The principal example occurs when $\mathbf{A} \in \mathbb{C}^{d \times d}$ and $\mathbf{A} = \mathbf{A}^*$, i.e. the matrix is Hermitian. In case $\mathbf{A} \in \mathbb{R}^{d \times d}$, this means that $\mathbf{A} = \mathbf{A}^T$ is squared and symmetric. In both above cases we have that

(Real) All eigenvalues $\{\lambda_i\}_{i=1}^d$ are real valued.

(Ortho) All eigenvectors are orthonormal to each other, or $\mathbf{x}_i^T\mathbf{x}_j = \delta_{i-j}$.

Such orthonormal matrices are often represented using the symbol $\mathbf{U}$, here for example we have that $\mathbf{X} = \mathbf{U}$. The last property means that $\mathbf{U}^T\mathbf{U} = \mathbf{I}_d$, where $\mathbf{I}_d = \operatorname{diag}(1, \ldots, 1) \in \{0, 1\}^{d \times d}$ is the identity matrix of dimension d. But it also implies that $\mathbf{U}\mathbf{U}^T = \mathbf{U}(\mathbf{U}^T\mathbf{U})\mathbf{U}^T = (\mathbf{U}\mathbf{U}^T)^2$. Then, the only full-rank matrix $\mathbf{C} \in \mathbb{R}^{d \times d}$ which satisfies the problem $\mathbf{C}\mathbf{C} = \mathbf{C}$ is $\mathbf{I}_d$, such that we have also that $\mathbf{U}\mathbf{U}^T = I_d$. As such we can write

$$\mathbf{U}^T\mathbf{A}\mathbf{U} = \Lambda. \tag{2.58}$$

That is, the matrix of eigenvectors of a symmetric matrix *diagonalizes* the matrix. The proof of the above facts are far from trivial, both w.r.t. existence of such eigenpairs as well as concerning the properties of the decomposition, and we refer e.g. to [9], Appendix A for more information and pointers to relevant literature. Then we define the concepts of definiteness of a matrix as follows.

Definition 5 (Positive Definite Matrices) *A square matrix $\mathbf{A} \in \mathbb{R}^{d \times d}$ is called Positive Definite (PD) iff one has for all vectors $\mathbf{x} \in \mathbb{R}^d$ that*

$$\mathbf{x}^T\mathbf{A}\mathbf{x} > 0. \tag{2.59}$$

A matrix $\mathbf{A} = \mathbf{A}^*$ is called Positive Semi-Definite (PSD) iff one has for all vectors $\mathbf{v} \in \mathbb{R}^d$ that

$$\mathbf{x}^T\mathbf{A}\mathbf{x} \geq 0. \tag{2.60}$$

The notation $\mathbf{A} \succeq 0$ denotes the $\mathbf{A}$ is PSD, and $\mathbf{A} \succ 0$ means that $\mathbf{A}$ is PD.

In the same vein, one defines negative definite, negative semi-definite and non-definite matrices. It turns out that such properties of a squared matrix captures quite well how different matrices behave in certain cases.

Lemma 2 (A PD Matrix) *A matrix $\mathbf{A} = \mathbf{A}^* \in \mathbb{C}^{d\times d}$ is positive definite if any of the following conditions hold:*

(i) If all eigenvalues $\{\lambda_i\}$ are strictly positive.

(ii) If there exist a matrix $\mathbf{C} \in \mathbb{R}^{n\times d}$ of rank $\operatorname{rank}(\mathbf{A})$ where

$$\mathbf{A} = \mathbf{C}^T\mathbf{C}. \tag{2.61}$$

(iii) If the determinant of any submatrix of $\mathbf{A}$ is larger than zero. A submatrix of $\mathbf{A}$ is obtained by deleting $k < d$ rows and corresponding columns of $\mathbf{A}$.

This decomposition does not only characterize the properties of a matrix, but is as well optimal in a certain sense.

Lemma 3 (Rayleigh Coefficient) *Let $\lambda_1 \geq \cdots \geq \lambda_d$ be the ordered eigenvalues of a matrix $\mathbf{A} = \mathbf{A}^T \in \mathbb{R}^{d\times d}$. Then*

$$\lambda_n = \min_{\mathbf{x}} \frac{\mathbf{x}^T\mathbf{A}\mathbf{x}}{\mathbf{x}^T\mathbf{x}}, \tag{2.62}$$

and this minimum is achieved when $\mathbf{x} \propto \mathbf{x}_1$, that is, is proportional to an eigenvector corresponding to a minimal eigenvalue.

$$\lambda_n = \max \min_{\mathbf{x}} \frac{\mathbf{x}^T\mathbf{A}\mathbf{x}}{\mathbf{x}^T\mathbf{x}}, \tag{2.63}$$

Moreover, one has for all $i = 1, \ldots, d$ that

$$\lambda_i = \max_{\mathbf{W}\in\mathbb{R}^{d\times(d-i)}} \min_{\mathbf{x}:\mathbf{W}^T\mathbf{x}=0} \frac{\mathbf{x}^T\mathbf{A}\mathbf{x}}{\mathbf{x}^T\mathbf{x}} = \min_{\mathbf{W}\in\mathbb{R}^{d\times(i-1)}} \max_{\mathbf{x}:\mathbf{W}^T\mathbf{x}=0} \frac{\mathbf{x}^T\mathbf{A}\mathbf{x}}{\mathbf{x}^T\mathbf{x}}, \tag{2.64}$$

which is known as the Courant-Fischer-Weyl min-max principle.

This is intuitively seen as

$$\lambda_i = \mathbf{x}_i^T\mathbf{A}\mathbf{x}_i = \frac{\mathbf{x}_i^T\mathbf{A}\mathbf{x}_i}{\mathbf{x}_i^T\mathbf{x}_i},\ \mathbf{x}_i\mathbf{x}_j^T,\ \forall j \neq i, \tag{2.65}$$

for all $i = 1, \ldots, d$, by definition of an eigenpair. Equation (2.64) implies that the eigenvectors are also endowed with an optimality property.

2.3.3 Singular Value Decompositions

While the EVD is usually used in case $\mathbf{A} = \mathbf{A}^*$ is Hermitian and PSD, the related Singular Vector Decomposition is used when $\mathbf{A} \in \mathbb{C}^{n\times d}$ is rectangular.

Definition 6 (Singular Value Decomposition) *Given a matrix $\mathbf{A} \in \mathbb{C}^{n\times d}$, the Singular Value Decomposition (SVD) is given as*

$$\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^*, \tag{2.66}$$

where $\mathbf{U} = (\mathbf{u}_1, \ldots, \mathbf{u}_n) \in \mathbb{C}^{n\times n}$ and $\mathbf{V} = (\mathbf{v}_1, \ldots, \mathbf{v}_d) \in \mathbb{C}^{d\times d}$ are both unitary matrices, such that $\mathbf{U}^T\mathbf{U} = \mathbf{U}\mathbf{U}^T = \mathbf{I}_n$ and $\mathbf{V}^T\mathbf{V} = \mathbf{V}\mathbf{V}^T = \mathbf{I}_d$. The matrix $\Sigma \in \mathbb{R}^{n\times d}$ which is all zero except for the elements $\Sigma_{ii} = \sigma_i$ for $i = 1, \ldots, \min(n, d)$. Here, $\{\sigma_i\}$ are the singular vectors, and the corresponding vectors $\{\mathbf{u}_i\} \subset \mathbb{C}^n$ and $\{\mathbf{v}_i\} \subset \mathbb{C}^d$ are called the left- and right singular vectors respectively.

The fundamental result then goes ass follows.

Lemma 4 (Existence and Uniqueness) *Given a matrix $\mathbf{A} \in \mathbb{C}^{n\times d}$, the Singular Value Decomposition (SVD) always exists and is unique up to linear transformations of the singular vectors corresponding to equal singular values.*

This implies that

$$\operatorname{rank}(\mathbf{A}) = |\{\sigma_i \neq 0\}|, \tag{2.67}$$

that is, the rank of a matrix equals the number of nonzero singular values of that matrix. The intuition behind this result is that the transformations $\mathbf{U}$ and $\mathbf{V}$ do not change the rank of a matrix, and the rank of Σ equals by definition the number of non-zero 'diagonal' elements. Similarly, the 'best' rank r approximation of a matrix $\mathbf{A}$ can be computed explicitly in terms of the SVD. Formally,

$$\mathbf{B} = \underset{\mathbf{B}\in\mathbb{C}^{n\times d}}{\operatorname{argmin}} \|\mathbf{A} - \mathbf{B}\|_F \quad \text{s.t.} \quad \operatorname{rank}(\mathbf{B}) = r. \tag{2.68}$$

where the *Frobenius* norm of a matrix $\mathbf{A}$ is defined as $\|\mathbf{A}\|_F = \operatorname{tr}(\mathbf{A}^T\mathbf{A})$. For simplicity, assume that the singular values which are not equal to zero are distinct, and sort them as $\sigma_{(1)} > \ldots \sigma_{(d')} \geq 0$ where $\min(n, d) \geq d' > r$. This notation is often used: $a_1, \ldots, a_n$ denoted a sequence of numbers, and $a_{(1)}, \ldots, a_{(n)}$ denotes the corresponding sorted sequence of numbers. The unique matrix optimizing this problem is given as

$$\hat{\mathbf{B}} = \sum_{i=1}^{r} \sigma_{(i)}\mathbf{u}_{(i)}\mathbf{v}_{(i)}^*. \tag{2.69}$$

where $\mathbf{u}_{(i)}, \mathbf{v}_{(i)}$ are the left- and right singular vector corresponding to eigenvalue $\sigma_{(i)}$. In matrix notation this becomes

$$\hat{\mathbf{B}} = \begin{bmatrix}\mathbf{U}_1 & \mathbf{U}_2\end{bmatrix} \begin{bmatrix}\Sigma^r & 0\\ 0 & 0\end{bmatrix} \begin{bmatrix}\mathbf{V}_1^*\\ \mathbf{V}_1^*\end{bmatrix} = \mathbf{U}\Sigma_{(r)}\mathbf{V}^*, \tag{2.70}$$

where Σ^r denote the matrix consisting of the first r rows and columns of Σ, and $\Sigma_{(r)} \in \mathbb{R}^{n\times d}$ equals Σ except for the singular values $\sigma_{(r+1)}, \sigma_{(r+2)}, \ldots$ which are set to zero. This result appeals again to the min-max result of the EVD. That is, the EVD and SVD decomposition are related as

Proposition 1 (SVD - EVD) *Let $\mathbf{A} \in \mathbb{C}^{n\times d}$, let then $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^*$ be the SVD, then*

$$\mathbf{A}^T\mathbf{A} = \mathbf{V}^*\Sigma^T\mathbf{U}\mathbf{U}^T\Sigma\mathbf{V} = \mathbf{V}^*(\Sigma^T\Sigma)\mathbf{V}. \tag{2.71}$$

Let $\mathbf{XA} = \Lambda\mathbf{X}$ *be the EVD of the PSD Hermitian matrix* $\mathbf{A}^T\mathbf{A}$*. Then* $\Sigma^T\Sigma = \Lambda$ *and* $\mathbf{X} = \mathbf{V}$*. That is* $\sigma_i^2 = \lambda_i$ *for all* $i = 1, \ldots, \min(d, n)$ *and* $\lambda_i = 0$ *otherwise. Similarly,*

$$\mathbf{A}\mathbf{A}^T = \mathbf{U}\Sigma\mathbf{V}^*\mathbf{V}\Sigma^T\mathbf{U}^* = \mathbf{U}(\Sigma\Sigma^T)\mathbf{U}. \tag{2.72}$$

and $\mathbf{V}$ *as such contains the eigenvectors of the outer-product* $\mathbf{A}\mathbf{A}^T$*, and the subsequent eigenvalues are* $\lambda_i = \sigma_i^2$ *for all* $i = 1, \ldots, \min(d, n)$ *and* $\lambda_i = 0$ *otherwise*

2.3.4 Other Matrix Decompositions

There exist a plethora of other matrix decompositions, each with its own properties. For the sake of this course the QR-decomposition is given. Let $\mathbf{A} = \mathbf{A}^T \in \mathbb{R}^{d\times d}$ be a symmetric positive definite matrix (i.e. without zero singular values). Then we can decompose the matrix $\mathbf{A}$ uniquely as the product of an uppertriangular matrix $\mathbf{R} \in \mathbb{R}^{d\times d}$ and a unitary matrix $\mathbf{Q} \in \mathbb{R}^{d\times d}$ or

$$\mathbf{A} = \begin{bmatrix} q_{11} & \ldots & q_{1d} \\ \vdots & & \vdots \\ q_{d1} & \ldots & q_{dd} \end{bmatrix} \begin{bmatrix} r_{11} & \ldots & r_{1d} \\ & \ddots & \vdots \\ 0 & & r_{dd} \end{bmatrix} = \mathbf{QR}, \tag{2.73}$$

where $\mathbf{Q}\mathbf{Q}^T = \mathbf{Q}^T\mathbf{Q} = \mathbf{I}_d$.

2.3.5 Indirect Methods

Let us return to the question how to solve the normal equations (2.42) associated to a least squares problem, given as

$$(\mathbf{X}^T\mathbf{X})\theta = \mathbf{X}^T\mathbf{y}. \tag{2.74}$$

Rather than solving the normal equations directly by using numerical techniques, one often resorts to solving related systems. For example in order to achieve numerical stable results, to speed up multiple estimation problems. Some common approaches go as follows

QR: If $\mathbf{A}\theta = \mathbf{b}$ is a set of linear equations where $\mathbf{A}$ is upper-triangular, then the solution θ can be found using a simple backwards substitution algorithm. But the normal equations can be phrased in this form using the QR decomposition of the covariance matrix as $(\mathbf{X}^T\mathbf{X}) = \mathbf{QR}$ where $\mathbf{Q}$ is orthonormal (unitary) and $\mathbf{R}$ is upper-triangular. Hence

$$(\mathbf{X}^T\mathbf{X})\theta = (\mathbf{X}^T\mathbf{y}) \Leftrightarrow \mathbf{Q}^T(\mathbf{X}^T\mathbf{X})\theta = \mathbf{Q}^T(\mathbf{X}^T\mathbf{y}) \Leftrightarrow \mathbf{R}\theta = \mathbf{b}, \tag{2.75}$$

where $\mathbf{b} = \mathbf{Q}^T(\mathbf{X}^T\mathbf{y})$. Hence the solution θ can then be found by backwards substitution. The QR decomposition of the matrix $(\mathbf{X}^T\mathbf{X})$ can be found using a Gram-Schmid algorithm, or using Householder or Givens rotations. Such approaches have excellent numerical robustness properties.

SVD: Given the SVD of the matrix $\mathbf{X} \in \mathbb{R}^{n\times d}$ as $\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^T$, and assume that all the singular values $\{\sigma_i > 0\}$ are strictly positive. Then the solution θ_n to (2.74) is given as

$$(\mathbf{X}^T\mathbf{X})\theta = (\mathbf{X}^T\mathbf{y}) \Leftrightarrow (\mathbf{V}\Sigma^T\mathbf{U}^T\mathbf{U}\Sigma\mathbf{V}^T)\theta = \mathbf{V}\Sigma^T\mathbf{U}^T\mathbf{y} \Leftrightarrow \theta = \mathbf{V}\Sigma^{-1}\mathbf{U}^T\mathbf{y}, \tag{2.76}$$

where $\Sigma^{-1} = \text{diag}(\sigma_1^{-1}, \ldots, \sigma_d^{-1}) \in \mathbb{R}^{d\times d}$ and the inverses exist by assumption.

†: In case the matrix $\mathbf{X}$ is not full rank, one has to modify the reasoning somewhat. That is, it is not guaranteed that there exists a solution θ_n to the normal equations of eq. (2.74). And in case that a solution θ_n exists, it will not be unique: assume that $\mathbf{a} \in \mathbb{R}^d$ is a nonzero vector such that $\mathbf{Xa} = 0$, then $\theta_n + \mathbf{a}$ solves the normal equations as well as

$$(\mathbf{X}^T\mathbf{X})(\theta_n + a_n) = (\mathbf{X}^T\mathbf{X})\theta_n = \mathbf{X}^T\mathbf{y}. \tag{2.77}$$

So it makes sense in case $\mathbf{X}$ is rank-deficient to look for a solution θ_n which is solves the normal equations as good as possible, while taking the lowest norm of all equivalent solutions. From properties of the SVD we have that any vector $\theta \in \mathbb{R}^d$ solving the problem as well as possible is given as

$$\theta = \sum_{i=1}^{r} \mathbf{v}_{(i)}\sigma_{(i)}^{-1}\mathbf{u}_{(i)}^T\mathbf{y} + \sum_{j=1}^{d-r} a_j\mathbf{v}_{(r+j)}a_j\mathbf{u}_{(r+j)}^T\mathbf{y}, \tag{2.78}$$

where $\{\sigma_{(1)}, \ldots, \sigma_{(r)}\}$ denote the r non-zero singular values. The smallest solution θ in this set is obviously the one where $a_1 = \cdots = a_{d-r} = 0$, or

$$\theta_n = \sum_{i=1}^{r} \mathbf{v}_{(i)}\sigma_{(i)}^{-1}\mathbf{u}_{(i)}^T\mathbf{y}. \tag{2.79}$$

Note that this is not quite the same as the motivation behind ridge regression where we want to find the solution trading the smallest norm requirement with the least squares objective.

From a practical perspective, the last technique is often used in order to get the best numerically stable technique. In common software packages as MATLAB, solving of the normal equations can be done using different commands. The most naive one is as follows:

```
>> theta = inv(X'*X) * (X'y)
```

But since this requires the involved inversion of a square matrix, a better approach is

```
>> theta = (X'*X) \ (X'y)
```

which solves the set of normal equations. This approach is also to be depreciated as it requires the software to compute the matrix $(\mathbf{X}^T\mathbf{X})$ explicitly, introducing numerical issues as a matrix-matrix product is known to increase rounding errors. The better way is

```
>> theta = pinv(X)*Y
```

MATLAB implements such technique using the shorthand notation

```
>> theta = X \ Y
```

2.4 Orthogonal Projections

Let us put our geometric glasses on, and consider the calculation with vectors and vector spaces. A vector space $\mathcal{A} \subset \mathbb{R}^m$ generated by a matrix $\mathbf{A} \in \mathbb{R}^{n\times m}$ is defined as

$$\mathcal{A} = \left\{ \mathbf{a} \;\middle|\; \mathbf{a} = \sum_{j=1}^{m} \mathbf{w}_j\mathbf{A}^j = \mathbf{Aw}, \forall \mathbf{w} \in \mathbb{R}^m \right\}. \tag{2.80}$$

Consider the following geometric problem: "Given a vector $\mathbf{x} \in \mathbb{R}^n$, and a linear space $\mathcal{A}$, extract from $\mathbf{x}$ the contribution lying in $\mathcal{A}$." Mathematically, this question is phrased as a vector which can be written as $\hat{\mathbf{x}} = \mathbf{A}\mathbf{w}$, where

$$(\mathbf{x} - \mathbf{A}\mathbf{w})^T \mathbf{A} = 0, \tag{2.81}$$

saying that "the remainder $\mathbf{x} - \hat{\mathbf{x}}$ cannot contain any component that correlate with $\mathbf{A}$ any longer". The projection $\mathbf{A}\mathbf{w}$ for this solution becomes as such

$$\mathbf{A}\mathbf{w} = \mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}\mathbf{x}, \tag{2.82}$$

that is, if the matrix $(\mathbf{A}^T\mathbf{A})$ can be inverted. Yet in other words, we can write that the projection $\hat{\mathbf{x}}$ of the vector $\mathbf{x}$ onto the space spanned by the matrix $\mathbf{A}$ can be written as

$$\hat{\mathbf{x}} = \Pi_{\mathbf{A}}\mathbf{x} = \mathbf{A}\left((\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}\mathbf{x}\right) = (\mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A})\mathbf{x}, \tag{2.83}$$

and the matrix $\Pi_{\mathbf{A}} = (\mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A})$ is called the projection matrix. Examples are

- The identity projection $\Pi_{\mathbf{A}} = I_n$, projecting any vector on itself.
- The coordinate projection $\Pi_{\mathbf{A}} = \text{diag}(1, 0, \ldots, 0)$, projecting any vector onto its first coordinate.
- Let $\Pi_{\mathbf{w}} = \frac{1}{\mathbf{w}^T\mathbf{w}}(\mathbf{w}\mathbf{w}^T)$ for any nonzero vector $\mathbf{w} \in \mathbb{R}^n$, then $\Pi_{\mathbf{w}}$ projects any vector orthogonal onto $\mathbf{w}$.
- In general, since we have to have that $\Pi_{\mathbf{A}}\Pi_{\mathbf{A}}\mathbf{x} = \Pi_{\mathbf{A}}\mathbf{x}$ for all $\mathbf{x} \in \mathbb{R}^n$ (idempotent property), a projection matrix $\Pi_{\mathbf{A}}$ has eigenvalues either 1 or zero.

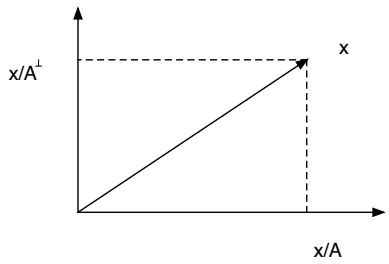


Figure 2.4: Orthogonal projection of vector $\mathbf{x}$ on the space $\mathcal{A}$, spanned by vector $\mathbf{a}$;

2.4.1 Principal Component Analysis

Principal Component Analysis (PCA) aims at uncovering the structure hidden in data. Specifically, given a number of samples $\mathcal{D} = \{\mathbf{x}_i\}_{i=1}^n$ - each $\mathbf{x}_i \in \mathbb{R}^d$ - PCA tries to come up with a shorter description of this set using less than d features. In that sense, it tries to 'compress' data, but it turns out that this very method shows up using other motivations as well. It is unlike a Least Squares estimate as there is no reference to a label or an output, and it is sometimes referred to as an *unsupervised* technique, [**?**]. The aim is translated mathematically as finding that direction

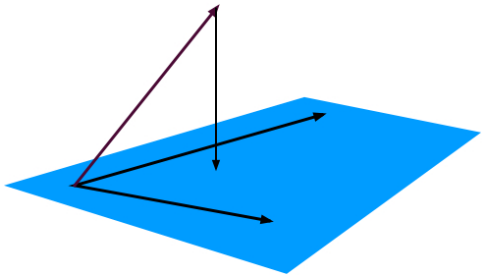


Figure 2.5: Schematic Illustration of an orthogonal projection of the angled upwards directed vector on the plane spanned by the two vectors in the horizontal plane.

$\mathbf{w} \in \mathbb{R}^d$ that explains most of the variance of the given data. 'Explains variance' of a vector is encoded as the criterion $\sum_{i=1}^n (\mathbf{x}_i^T \mathbf{w})^2$. Note that multiplication of the norm of the vector $\mathbf{w}$ gives a proportional gain in the 'explained variance'. As such it makes sense to fix $\|\mathbf{w}\|_2 = 1$, or $\mathbf{w}^T\mathbf{w} = 1$, in order to avoid that we have to deal with infinite values.

This problem is formalized as the following optimization problem. Let $\mathbf{x}_i \in \mathbb{R}^d$ be the observation made at instant i, and let $\mathbf{w} \in \mathbb{R}^d$ and let $z_i \in \mathbb{R}$ be the latent value representing $\mathbf{x}_i$ in a one-dimensional subspace. Then the problem becomes

$$\min_{\mathbf{w}\in\mathbb{R}^d, \{\mathbf{z}_i\}_i} \sum_{i=1}^n \|\mathbf{x}_i - \mathbf{w}\mathbf{z}_i\|_2^2 \quad \text{s.t.} \quad \mathbf{w}^T\mathbf{w} = 1. \tag{2.84}$$

In order to work out how to solve this optimization problem, let us again define the matrix $\mathbf{X}_n \in \mathbb{R}^{n\times d}$ stacking up all the observations in $\mathcal{D}$, and the matrix $\mathbf{z}_n \in \mathbb{R}^n$ stacking up all the corresponding latent values, or

$$\mathbf{X}_n = \begin{bmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \\ \vdots \\ \mathbf{x}_n^T \end{bmatrix}, \quad \mathbf{z}_n = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{bmatrix}. \tag{2.85}$$

Then the problem eq. (2.84) can be rewritten as

$$\min_{\mathbf{w}\in\mathbb{R}^d, \mathbf{z}_n\in\mathbb{R}^n} J_n(\mathbf{z}_n, \mathbf{w}) = \operatorname{tr}\left(\mathbf{X}_n - \mathbf{z}_n\mathbf{w}^T\right)\left(\mathbf{X}_n - \mathbf{z}_n\mathbf{w}^T\right)^T \quad \text{s.t.} \quad \mathbf{w}^T\mathbf{w} = 1, \tag{2.86}$$

where $\operatorname{tr}\mathbf{G} = \sum_{i=1}^n \mathbf{G}_{ii}$ where $\mathbf{G} \in \mathbb{R}^{n\times n}$. Suppose that $\mathbf{w}$ satisfying $\mathbf{w}^T\mathbf{w} = 1$ were known, then we can find the corresponding optimal $\hat{\mathbf{z}}_n(\mathbf{w})$ as a simple least squares problem: as classically we derive the objective to eq. (2.86) and equate it to zero, giving the condition for any $i = 1, \ldots, n$ that

$$\frac{\partial J_n(\hat{\mathbf{z}}_{n,i}(\mathbf{w}), \mathbf{w})}{\partial \hat{\mathbf{z}}_{n,i}(\mathbf{w})} = 0 \Leftrightarrow -2\sum_{i=1}^n \left(\mathbf{x}_i - \mathbf{w}\hat{\mathbf{z}}_{n,i}(\mathbf{w})\right)^T \mathbf{w} = 0, \tag{2.87}$$

or

$$\hat{\mathbf{z}}_n(\mathbf{w}) = \frac{1}{(\mathbf{w}^T\mathbf{w})}\mathbf{X}_n\mathbf{w}. \tag{2.88}$$

Now having this closed form solution for $\hat{\mathbf{z}}_n$ corresponding to a $\mathbf{w}$, one may invert the reasoning and try to find this $\mathbf{w}$ satisfying the constraint $\mathbf{w}^T\mathbf{w} = 1$ and optimizing the objective $J_n(\hat{\mathbf{z}}_n(\mathbf{w}), \mathbf{w})$ as

$$\min_{\mathbf{w}\in\mathbb{R}^d} J'_n(\mathbf{w}) = J_n(\hat{\mathbf{z}}_n(\mathbf{w}), \mathbf{w}) = \operatorname{tr}\left(\mathbf{X}_n - \hat{\mathbf{z}}_n(\mathbf{w})\mathbf{w}^T\right)\left(\mathbf{X}_n - \hat{\mathbf{z}}_n(\mathbf{w})\mathbf{w}^T\right)^T. \quad \text{s.t.} \quad \mathbf{w}^T\mathbf{w} = 1. \tag{2.89}$$

Working out the terms and using the normal equations (2.88) gives the equivalent optimization problem

$$\min_{\mathbf{w}\in\mathbb{R}^d} J'_n(\mathbf{w}) = \left\|\mathbf{X}_n - (\mathbf{w}\mathbf{w}^T)\mathbf{X}_n\right\|_F \quad \text{s.t.} \quad \mathbf{w}^T\mathbf{w} = 1. \tag{2.90}$$

where the Frobenius norm $\|\cdot\|_F^2$ is defined for any matrix $\mathbf{G} \in \mathbb{R}^{n\times d}$ as

$$\|\mathbf{G}\|_F = \operatorname{tr}\mathbf{G}\mathbf{G}^T = \sum_{i=1}^{n} \mathbf{G}_i^T\mathbf{G}_i = \sum_{ij} \mathbf{G}_{ij}^2, \tag{2.91}$$

and where $\mathbf{G}_i$ denotes the ith row of $\mathbf{G}$. It is useful to interpret this formula. It is easy to see that the matrix $\Pi_{\mathbf{w}} = (\mathbf{w}\mathbf{w}^T)$ as a projection matrix as $(\mathbf{w}^T\mathbf{w})^{-1} = 1$ by construction, and as such we look for the best projection such that $\Pi\mathbf{X}_n$ is as close as possible to $\mathbf{X}_n$ using a Euclidean norm. To solve this optimization problem, let us rewrite eq. (2.90) in terms of the arbitrary vector $\mathbf{v} \in \mathbb{R}^d$ such that $\mathbf{w} = \frac{\mathbf{v}}{\|\mathbf{v}\|_2}$ has norm 1 by construction. We take care of this rescaling by dividing the objective through $\mathbf{v}^T\mathbf{v}$. Recall that linearity of the trace implies $\operatorname{tr}\mathbf{G}\mathbf{G}^T = \operatorname{tr}\mathbf{G}^T\mathbf{G}$. Since $(\mathbf{w}\mathbf{w})^T(\mathbf{w}\mathbf{w}) = (\mathbf{w}\mathbf{w})$ (idempotent), one has

$$\min_{\mathbf{v}\in\mathbb{R}^d} J'_n(\mathbf{v}) = \min_{\mathbf{v}\in\mathbb{R}^d} \frac{\mathbf{v}^T\mathbf{v} - \mathbf{v}^T(\mathbf{X}_n^T\mathbf{X}_n)\mathbf{v}}{\mathbf{v}^T\mathbf{v}} = 1 - \max_{\mathbf{v}\in\mathbb{R}^d} \frac{\mathbf{v}^T(\mathbf{X}_n^T\mathbf{X}_n)\mathbf{v}}{\mathbf{v}^T\mathbf{v}}, \tag{2.92}$$

and $\mathbf{w}$ solving eq. (2.84) is given as $\mathbf{w} = \frac{\mathbf{v}}{\|\mathbf{v}\|}$.

Now luckily enough maximization of $\frac{\mathbf{v}^T(\mathbf{X}_n^T\mathbf{X}_n)\mathbf{v}}{\mathbf{v}^T\mathbf{v}}$ is a wellknown problem, studied for decades in analyses and numerical algebra as the problem of maximizing the Rayleigh coefficient. From this we know not only how the maximum is found, but how all local maxima can be found. Equating the derivative of the Rayleigh coefficient to zero gives the conditions

$$\lambda(\mathbf{v}) = \frac{\mathbf{v}^T(\mathbf{X}_n^T\mathbf{X}_n)\mathbf{v}}{\mathbf{v}^T\mathbf{v}} \Leftrightarrow \lambda(\mathbf{v})(\mathbf{v}^T\mathbf{v}) = \mathbf{v}^T(\mathbf{X}_n^T\mathbf{X}_n)\mathbf{v}. \tag{2.93}$$

Now deriving to $\mathbf{v}$ and equating to zero gives the conditions

$$\lambda(\mathbf{v})\mathbf{v} = (\mathbf{X}_n^T\mathbf{X}_n)\mathbf{v}, \tag{2.94}$$

and we know that the d orthogonal solutions $\{\mathbf{v}_i\}_i$ and corresponding coefficients $\{\lambda(\mathbf{v}_i)\}$ are given by the eigenvectors and eigenvalues of the matrix $\mathbf{X}_n^T\mathbf{X}_n$, such that $\mathbf{v}_i^T\mathbf{v}_j = \delta_{ij}$ (i.e. is one if $i = j$, and zero otherwise), and $\mathbf{v}_i^T(\mathbf{X}_n^T\mathbf{X}_n)\mathbf{v}_j = \lambda_i(\mathbf{v}_i)\delta_{ij}$. We will use the notation that $\{(\lambda_i(\mathbf{X}_n^T\mathbf{X}_n), \mathbf{v}_i(\mathbf{X}_n^T\mathbf{X}_n))\}$ to denote this set. In fact, the relation PCA - eigenvalue decomposition is so close that they are often considered to be one and the same. That is if an algorithm performs an eigenvalue decomposition at a certain stage of a certain matrix, one may often think of it as a PCA of this matrix thereby helping intuition.

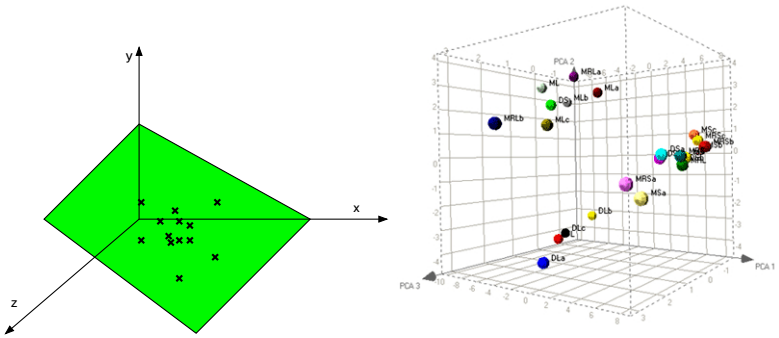


Figure 2.6: (a) An example of $n = 13$ and $d = 2$, where all the samples 'x' lie in a two-dimensional linear subspace denoted as the filled rectangle. PCA can be used to recover this subspace from the data matrix $\mathbf{X} \in \mathbb{R}^{13\times 3}$. (b) An example of the results of a PCA analysis on 2000 of expression levels observed in 23 experiments. The 3 axes correspond with the 3 principal components of the matrix $\mathbf{X} \in \mathbb{R}^{23\times 2000}$.