ECE 7750: Mathematical Foundations of Machine Learning **Fall 2022**

Lecture 0: Supplementary note on finite dimensional least squares

Lecturer: Ashwin Pananjady

This note is meant as supplementary material to jog your memory on finite-dimensional linear systems and least squares. It will make use of several tools (like gradients, inner products, etc.) in finite dimensional spaces. Some of the notation here is slightly different from what we've been using in the Hilbert spaces component of the class (and follows the $\boldsymbol{x} \to y$ supervised learning notation in ML), but this should hopefully not be a problem since everything in this note is self-contained.

0.1. Warming up

The opening act of our story today will be systems of linear equations. Suppose for the moment that we have a $d \times d$ matrix $\boldsymbol{A}$ containing real values, and the system of equations

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{w} \tag{0.1}$$

for a known vector $\boldsymbol{y} \in \mathbb{R}^d$ and an unknown vector $\boldsymbol{w} \in \mathbb{R}^d$. The above equation can be written as a collection of d distinct equations (In general, don't hesitate to write out vector equations like these explicitly as build intuition for manipulating them.)

$$y_i = \sum_{j=1}^{d} A_{i,j} w_j \text{ for each } i = 1, \ldots, d.$$

Another way to write this system of linear equations is to notice that $\boldsymbol{A}\boldsymbol{w} = \sum_{j=1}^{d} A_j w_j$, where A_j is the j-th column of the matrix $\boldsymbol{A}$, so that we are looking for a solution to the system

$$\boldsymbol{y} = \sum_{j=1}^{d} A_j w_j.$$

When does the linear system (0.1) have a solution $\boldsymbol{w}$, and if so, when is this solution unique? Your linear algebra class would have told you that the answer is "it depends". In general, there will be three cases:

- $\boldsymbol{A}$ has full column rank: This is just another way of saying that the vectors forming the columns of $\boldsymbol{A}$ *span* the entire space $\mathbb{R}^d$, i.e., the range of $\boldsymbol{A}$ is all of $\mathbb{R}^d$. In particular, *any* d-dimensional vector $\boldsymbol{u} \in \mathbb{R}^d$ can be written in the form $\boldsymbol{u} = \sum_{j=1}^{d} \alpha_j A_j$, where $\alpha_1, \ldots, \alpha_d$ are some real coefficients. In particular, this means that the vector $\boldsymbol{y}$ can

be written in this form, and the corresponding coefficients $\alpha_j, j = 1, \ldots, d$ can be collected in the vector $\boldsymbol{w}$. Is the resulting vector unique? The answer is yes, since a square matrix with full column rank is actually *invertible*, so that we can explicitly write $\boldsymbol{w} = \boldsymbol{A}^{-1}\boldsymbol{y}$.

- $\boldsymbol{A}$ does not have full column rank and $\boldsymbol{y}$ is not in the span of $A_1, \ldots, A_d$. In this case, we know that the column vectors $A_1, \ldots, A_d$ are *linearly* dependent, and do not span all of d-dimensional space. For a simple example with $d = 2$:

  Example 1 *Take* $\boldsymbol{y} = (2, 1)$ *and the matrix* $\boldsymbol{A} = \begin{bmatrix} 1 & 0.5 \\ 0 & 0 \end{bmatrix}$. *Any linear combination of the columns of* $\boldsymbol{A}$ *will only have non-zeros in the first entry, i.e., will be a vector of the form* $(\alpha, 0)$. *All such vectors only form a* linear subpace *of dimension* 1*, and in particular, since* $\boldsymbol{y}$ *has a nonzero in its second entry, it cannot be written as* $\boldsymbol{Aw}$ *for any* $\boldsymbol{w}$. *A reasonable (but different) problem in such cases is to try to find the* closest *point to* $\boldsymbol{y}$ *in the range of* $\boldsymbol{A}$*; more on this later.*

- $\boldsymbol{A}$ does not have full column rank, but $\boldsymbol{y}$ lies in the range of $\boldsymbol{A}$: By definition, we can write $\boldsymbol{y} = \boldsymbol{Aw}$ for some $\boldsymbol{w}$, since $\boldsymbol{y}$ lies in the range of $\boldsymbol{A}$. However, this $\boldsymbol{w}$ is not unique. To see this, let us use a variant of the previous example:

  Example 2 *Take* $\boldsymbol{y} = (2, 0)$ *and the matrix* $\boldsymbol{A} = \begin{bmatrix} 1 & 0.5 \\ 0 & 0 \end{bmatrix}$. *Then* $\boldsymbol{w} = (2, 0)$ *is a solution, but so is* $\boldsymbol{w} = (1/2, 1)$. *Indeed, any vector of the form* $\boldsymbol{w} = (2 + \alpha, -2\alpha))$ *is a solution.*

  The reason for nonuniqueness in this example is that $\boldsymbol{A}$ has a *nullspace*, or kernel. This is the family of vectors $\mathsf{ker}(\boldsymbol{A}) = \{\boldsymbol{u} : \boldsymbol{Au} = \boldsymbol{0}\}$, which is also a linear subspace. In the above example, it is the set of all vectors of the form $\alpha \cdot (1, -2)$, which is a one-dimensional subspace. Why is the nullspace relevant to uniqueness? Take any solution $\boldsymbol{w}$ to $\boldsymbol{y} = \boldsymbol{Aw}$, and a vector $\boldsymbol{u} \in \mathsf{ker}(\boldsymbol{A})$. Then for any value of α, we have

  $$\boldsymbol{A}(\boldsymbol{w} + \alpha\boldsymbol{u}) = \boldsymbol{Aw} + \boldsymbol{0} = \boldsymbol{Aw},$$

  so that the $\boldsymbol{w}' = \boldsymbol{w} + \alpha\boldsymbol{u}$ also satisfies $\boldsymbol{y} = \boldsymbol{Aw}'$ and yields an infinite family of solutions as we vary α.

Everything we did so far was for square matrices $\boldsymbol{A}$. But going forward, we will be interested in $n \times d$ matrices where $n \geq d$, so that we are trying to solve system (0.1) for $\boldsymbol{y} \in \mathbb{R}^n$ and $\boldsymbol{w} \in \mathbb{R}^d$. How does the story above change in this case? Suppose the matrix $\boldsymbol{A} \in \mathbb{R}^{n \times d}$ has full column rank, which means that the columns of $\boldsymbol{A}$ span a d-dimensional subspace and that $\boldsymbol{A}$ does not have a nontrivial kernel (besides the all-zero vector). Then there are two cases:

- $\boldsymbol{y}$ is in the range of $\boldsymbol{A}$: Once again, by definition we have that $\boldsymbol{y} = \boldsymbol{Aw}$ for some $\boldsymbol{w} \in \mathbb{R}^d$, but how does one show uniqueness? Here is a proof by contradiction:

  Suppose that there is another $\boldsymbol{w}' \neq \boldsymbol{w}$ satisfying $\boldsymbol{y} = \boldsymbol{Aw}'$. Then $\boldsymbol{A}(\boldsymbol{w} - \boldsymbol{w}') = \boldsymbol{y} - \boldsymbol{y} = \boldsymbol{0}$, so that $\boldsymbol{w} - \boldsymbol{w}' \in \mathsf{ker}(\boldsymbol{A})$. But this contradicts the fact that $\boldsymbol{A}$ has full column rank, so the original supposition must have been false.

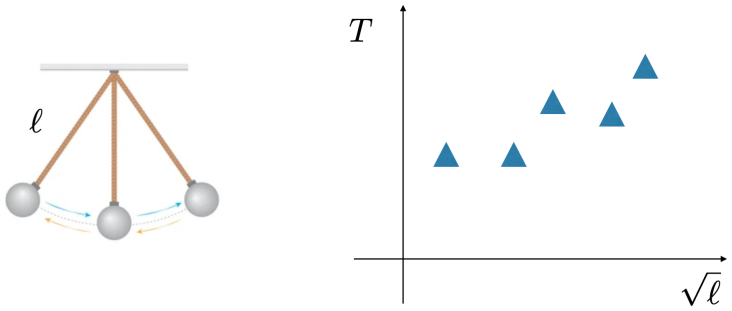

Figure 0.1: The pendulum example from the introductory lecture, in which we measure the time periods of pendula with varying lengths and plot these as a function of the root-length.

- $\boldsymbol{y}$ is not in the range of $\boldsymbol{A}$: In this case (and once again, by definition), we cannot express $\boldsymbol{y}$ in terms of a linear combination of the columns of $\boldsymbol{A}$. But this brings us to the main act of today: finding a solution $\boldsymbol{w}$ such that $\boldsymbol{y}$ is *close* to $\boldsymbol{Aw}$ in a certain sense.

0.2. From linear equations to linear regression

In the introductory lecture, we saw a toy example of linear regression, given by the problem of fitting the time period of a simple pendulum as a function of various measurements of its length. This example is recalled below:

Example 3 *Suppose we have n observations of the process illustrated in Figure 0.1. We measure the time period T_i of a simple pendulum having root-length $\sqrt{\ell_i}$; we are taking square roots here to reflect that we understand that the true time period ought to be directly proportional to the squared length. We execute this for various values of the length, and obtain observations $(T_i, \sqrt{\ell_i})_{i=1}^n$.*

Imagine for the moment that we posit a *linear model* for the relationship between T_i and $\sqrt{\ell_i}$. In this case, we would be "guessing" that $T_i = w_1\sqrt{\ell_i} + w_0$ for some scalars w_0 and w_1. To match notation with the previous section, let us collect the times T_i in a vector $\boldsymbol{y}$, and form the matrix

$$\boldsymbol{X} = \begin{bmatrix} \sqrt{\ell_1} & 1 \\ \sqrt{\ell_2} & 1 \\ \vdots & \vdots \\ \sqrt{\ell_n} & 1 \end{bmatrix}.$$

Then we are interested in finding a vector $\boldsymbol{w} = (w_1, w_2)$ such that $\boldsymbol{y} \approx \boldsymbol{Xw}$.

Note that as observed in class last time, one could very well have posited the *quadratic* model $T_i \approx w_2\ell_i + w_1\sqrt{\ell_i} + w_0$. This appears (on the face of it) to be drastically different,

but can still be solved like before. In particular, let $\boldsymbol{y}$ be the collection of T_i as before, let $\boldsymbol{w} = (w_2, w_1, w_0)$, and set

$$\boldsymbol{X} = \begin{bmatrix} \ell_1 & \sqrt{\ell_1} & 1 \\ \ell_2 & \sqrt{\ell_2} & 1 \\ \vdots & \vdots & \\ \ell_n & \sqrt{\ell_n} & 1 \end{bmatrix}.$$

From the discussion in the previous section, we will only obtain exact equality $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{w}$ if $\boldsymbol{y}$ is in the range of $\boldsymbol{X}$, i.e., all the observed points lie on a line in the first example, or on a quadratic curve in the second example. But this is clearly too much to hope for! What can we do in the absence of such a strong condition?

A reasonable thing to do (and we will justify assumptions under which this is reasonable later on) is to find the *closest* point to $\boldsymbol{y}$ in the range of $\boldsymbol{X}$, when measured in (squared) ℓ_2 norm. In other words, suppose we let $d = 2$ in the first case and $d = 3$ in the second case. Then we would like to find a point $\widehat{\boldsymbol{w}}$ such that

$$\widehat{\boldsymbol{w}} = \operatorname*{argmin}_{\boldsymbol{w} \in \mathbb{R}^d} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}\|_2^2 = \sum_{i=1}^{d} (y_i - \langle \boldsymbol{x}_i, \boldsymbol{w} \rangle)^2. \tag{0.2}$$

Here, we have used $\boldsymbol{x}_i$ to denote the i-th row of the matrix $\boldsymbol{X}$ (when viewed as a $d \times 1$ vector), and $\langle \cdot, \cdot \rangle$ once again denotes the inner product. Equation (0.2) is the basic *least squares* regression problem.

Let us highlight two key aspects of the examples you've seen so far:

- The *intercept* in the linear regression problem corresponds to the column of 1s in the matrix $\boldsymbol{X}$. Adding a column of ones will go from a model without intercepts to one that models intercepts.
- In going from the model depending only on $\sqrt{\ell}$ to one depending on ℓ, we exhibited the *lifting* trick, whereby one can take various functions of the feature in hand (ℓ in this case) and add them all on as features in the regression problem to form a longer vector $\boldsymbol{x}$. This will come back later when we study polynomial regression.

Before working towards the solution to the problem (0.2), let us recall why we are doing all this in the context of regression. The fundamental premise of regression is that we believe the approximate input-output relationship $\boldsymbol{y} \approx f(\boldsymbol{x})$, for some function $f \in \mathcal{F}$. In the above examples (after lifting to form $\boldsymbol{x}$), we are positing a *linear model class*, where any f in the class is given by the map $f(\boldsymbol{x}) = \langle \boldsymbol{x}, \boldsymbol{w} \rangle$ for some $\boldsymbol{w} \in \mathbb{R}^d$.

Having chosen the *square loss*, we are then interested in finding the best-fit linear function, which then yields the prediction $\boldsymbol{x} \mapsto \langle \boldsymbol{x}, \widehat{\boldsymbol{w}} \rangle$. This is an instance of working under some model class $\mathcal{F}$, and returning the specific $f \in \mathcal{F}$ that has the lowest *error* on our observed data points. In order to find a particular candidate model, we solve the optimization problem (0.2).

0.3. Solving the linear regression problem

Let us now turn to how we might solve for $\widehat{\boldsymbol{w}}$ in problem (0.2). Before we begin, we make a small change to notation so that we don't have to carry around cumbersome expressions

later on. We write $\boldsymbol{w} = (w_1, \ldots, w_d)$ and use $x_{i,j}$ to denote the j-th entry of the vector $\boldsymbol{x}_i$. In particular, we won't make distinctions between the intercept and remaining linear terms, since we can just add the all-ones column to the matrix and form a new $\boldsymbol{X}$ matrix.

First, note that the objective function in (0.2) is an "upward quadratic", and it will have a unique solution given by the point at which its gradient equals zero. We will unpack all this a bit more later on. Then one can compute the gradient of the function $\mathcal{L}(\boldsymbol{w}) = \sum_{i=1}^n (y_i - \langle \boldsymbol{x}_i, \boldsymbol{w} \rangle)^2$ by writing the individual partial derivatives. We have for each $j = 1, \ldots, d$ that

$$\frac{\partial \mathcal{L}}{\partial w_j} = 2 \sum_{i=1}^n (y_i - \langle \boldsymbol{x}_i, \boldsymbol{w} \rangle) \cdot (-x_{i,j}),$$

where we used the chain rule of differentiation. Collecting all of these (over all j) as a vector, we have

$$\nabla \mathcal{L}(\boldsymbol{w}) = 2 \sum_{i=1}^n (\langle \boldsymbol{x}_i, \boldsymbol{w} \rangle - y_i) \cdot \boldsymbol{x}_i = 2 \sum_{i=1}^n (\boldsymbol{x}_i \boldsymbol{x}_i^\top \boldsymbol{w} - \boldsymbol{x}_i y_i), \tag{0.3}$$

where in the last line we have used the fact that $\boldsymbol{x}_i^\top \boldsymbol{w}$ is a scalar, so that $\langle \boldsymbol{x}_i, \boldsymbol{w} \rangle \cdot \boldsymbol{x}_i = \boldsymbol{x}_i \cdot \langle \boldsymbol{x}_i, \boldsymbol{w} \rangle$. We can further simplify the RHS of Eq. (0.3) by noting that the sum over i does not involve $\boldsymbol{w}$, so

$$\nabla \mathcal{L}(\boldsymbol{w}) = 2 \left(\sum_{i=1}^n \boldsymbol{x}_i \boldsymbol{x}_i^\top \right) \boldsymbol{w} - 2 \sum_{i=1}^n \boldsymbol{x}_i y_i = 2(\boldsymbol{X}^\top \boldsymbol{X}) \boldsymbol{w} - 2 \boldsymbol{X}^\top \boldsymbol{y}.$$

The last step above uses the fact that $\sum_{i=1}^n \boldsymbol{x}_i \boldsymbol{x}_i^\top = \boldsymbol{X}^\top \boldsymbol{X}$ (convince yourself of this!), and that $\sum_{i=1}^n \boldsymbol{x}_i y_i = \boldsymbol{X}^\top \boldsymbol{y}$.

To conclude, note that the solution $\widehat{\boldsymbol{w}}$ must satisfy the $d \times d$ linear system $(\boldsymbol{X}^\top \boldsymbol{X}) \widehat{\boldsymbol{w}} = \boldsymbol{X}^\top \boldsymbol{y}$. Provided $\boldsymbol{X}^\top \boldsymbol{X}$ has full column rank, we then obtain

$$\widehat{\boldsymbol{w}} = (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}^\top \boldsymbol{y}. \tag{0.4}$$

Exercise 1 *Based on the discussion in the first section, think about what happens in this calculation if $\boldsymbol{X}^\top \boldsymbol{X}$ does not have full column rank.*

While we did a few lines of algebra to get to this point, it's important to remember the key takeaway: If the posited model class is linear and we use the squared loss, then the best-fit prediction to our data $(\boldsymbol{X}, \boldsymbol{y})$ can be computed in closed form, as above! In particular:

- For each point $\boldsymbol{x}_i$ in the dataset, we are returning the prediction $f(\boldsymbol{x}_i) = \langle \boldsymbol{x}_i, \widehat{\boldsymbol{w}} \rangle = \boldsymbol{x}_i^\top (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}^\top \boldsymbol{y}$. The vector of predictions over all n points is then given by

$$\widehat{\boldsymbol{y}} = \boldsymbol{X} (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}^\top \boldsymbol{y}. \tag{0.5}$$

- The linear function returned by this fit on any new point $\boldsymbol{x} \in \mathbb{R}^d$ is $f(\boldsymbol{x}) = \langle \boldsymbol{x}, \widehat{\boldsymbol{w}} \rangle = \boldsymbol{x}^\top (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}^\top \boldsymbol{y}$.

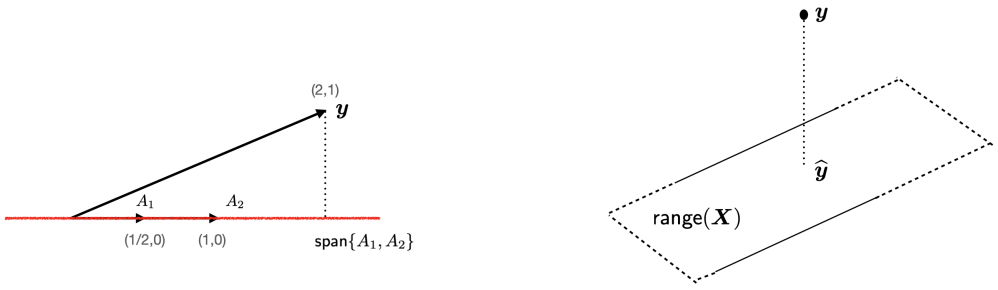

Figure 0.2: (Left) An illustration of the projection for Example 1. (Right) A schematic illustration in the general linear regression problem, where the vector $\boldsymbol{y} - \widehat{\boldsymbol{y}}$ is perpendicular to (i.e., has zero inner product with) any vector in the range of $\boldsymbol{X}$.

0.4. Geometry of least squares problems

We conclude with a discussion of some geometric aspects of the least squares problem, which will give you another way to visualize what is going on in finite dimensions.

0.4.1 Orthogonality principle

While the derivation above was analytical, one can also think of it in a geometric fashion. To build toward this, take Example 1, and suppose we are interested in finding the closest point to $\boldsymbol{y} = (2, 0)$ to the range of the matrix $\boldsymbol{A}$ (which is given by the linear subspace $\mathsf{span}\{(1, 0)\}$). Then geometrically, the closest point must be the *projection* of $\boldsymbol{y}$ onto this linear subspace, illustrated in Figure 0.2(Left). In particular, just looking at this picture, we have that $\boldsymbol{y} - \widehat{\boldsymbol{y}} \perp \mathsf{span}\{(1, 0)\}$.

How can we make similar intuition work in higher dimensions? It turns out that there is indeed a way to do so! Consider the residual vector $\boldsymbol{y} - \widehat{\boldsymbol{y}}$, where $\widehat{\boldsymbol{y}} = \boldsymbol{X}\widehat{\boldsymbol{w}}$ is the prediction on the data (0.5) obtained from the least squares fit. Then one can show that $\widehat{\boldsymbol{y}}$ is indeed the projection of $\boldsymbol{y}$ onto the range of $\boldsymbol{X}$. This is illustrated schematically in Figure 0.2(Right).

In order to formally establish this, we would like to show that $\boldsymbol{y} - \widehat{\boldsymbol{y}} \perp \boldsymbol{X}\boldsymbol{w}$ for all $\boldsymbol{w} \in \mathbb{R}^d$. To do so, write

$$\begin{aligned}(\boldsymbol{y} - \widehat{\boldsymbol{y}})^\top \boldsymbol{X}\boldsymbol{w} &= \boldsymbol{y}^\top \boldsymbol{X}\boldsymbol{w} - (\boldsymbol{X}(\boldsymbol{X}^\top\boldsymbol{X})^{-1}\boldsymbol{X}^\top\boldsymbol{y})^\top \boldsymbol{X}\boldsymbol{w} \\ &= \boldsymbol{y}^\top \boldsymbol{X}\boldsymbol{w} - \boldsymbol{y}^\top \boldsymbol{X}(\boldsymbol{X}^\top\boldsymbol{X})^{-1}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{w} \\ &= \boldsymbol{y}^\top \boldsymbol{X}\boldsymbol{w} - \boldsymbol{y}^\top \boldsymbol{X}\boldsymbol{w} = 0.\end{aligned}$$

In other words, the claimed result is indeed true! What is a consequence of this? For any $\boldsymbol{w} \in \mathbb{R}^d$, we have

$$\begin{aligned}\mathcal{L}(\boldsymbol{w}) = \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}\|_2^2 &= \|\boldsymbol{y} - \widehat{\boldsymbol{y}} + \boldsymbol{X}\widehat{\boldsymbol{w}} - \boldsymbol{X}\boldsymbol{w}\|_2^2 \\ &= \|\boldsymbol{y} - \widehat{\boldsymbol{y}}\|_2^2 + \|\boldsymbol{X}\widehat{\boldsymbol{w}} - \boldsymbol{X}\boldsymbol{w}\|_2^2 + 2\langle \boldsymbol{y} - \widehat{\boldsymbol{y}}, \boldsymbol{X}\widehat{\boldsymbol{w}} - \boldsymbol{X}\boldsymbol{w}\rangle \\ &= \|\boldsymbol{y} - \widehat{\boldsymbol{y}}\|_2^2 + \|\boldsymbol{X}(\boldsymbol{w} - \widehat{\boldsymbol{w}})\|_2^2,\end{aligned}$$

which confirms that $\mathcal{L}$ is an upward quadratic function.

0.4.2 Landscape of the function $\mathcal{L}$

Suppose now that we are interested in how one might *plot* $\mathcal{L}$ as a function of its argument $\boldsymbol{w}$, noting that this is complementary to the orthogonality question. Visualizing this function has a lot of applications; in particular, it will help us understand how iterative algorithms behave when they are run on such functions; indeed you visualized several such functions in HW1. Rewriting what we just had, we have

$$\mathcal{L}(\boldsymbol{w}) = \underbrace{\|\boldsymbol{y} - \widehat{\boldsymbol{y}}\|_2^2}_{\text{independent of } \boldsymbol{w}} + \|\boldsymbol{X}(\boldsymbol{w} - \widehat{\boldsymbol{w}})\|_2^2;$$

in other words this function is some constant independent of $\boldsymbol{w}$ added onto a quadratic function of $\boldsymbol{w}$. Let us look at the quadratic function $\mathcal{Q}(\boldsymbol{w}) = \|\boldsymbol{X}(\boldsymbol{w} - \widehat{\boldsymbol{w}})\|_2^2$ in more detail. Clearly, this is positive everywhere, and zero (i.e., minimized) when $\boldsymbol{w} = \widehat{\boldsymbol{w}}$. Generally speaking, we will see that it is an upward parabola, whose *contours* look like ellipsoids in $d-1$ dimensions centered at $\widehat{\boldsymbol{w}}$.

Let us present three examples below to provide some intuition. Note that we have $\mathcal{Q}(\boldsymbol{w}) = (\boldsymbol{w} - \widehat{\boldsymbol{w}})^\top \boldsymbol{X}^\top \boldsymbol{X}(\boldsymbol{w} - \widehat{\boldsymbol{w}})$. This representation makes it clear that all that matters here are the properties of the matrix $\boldsymbol{X}^\top \boldsymbol{X}$.

Example 4 *First, consider the case where we have $\boldsymbol{X}^\top \boldsymbol{X} = \boldsymbol{I}$, which happens when the columns of $\boldsymbol{X}$ are orthonormal vectors, so that $\langle X_i, X_j\rangle = \mathbf{1}(i = j)$ (once again X_i is the i-th column of $\boldsymbol{X}$). Here, the function just takes the form $\mathcal{Q}(\boldsymbol{w}) = \|\boldsymbol{w} - \widehat{\boldsymbol{w}}\|_2^2 = \sum_{i=1}^d (w_i - \widehat{w}_i)^2$. In other words, this is a vanilla quadratic function. If you looked at any of its contours, i.e., all points satisfying $\mathcal{Q}(\boldsymbol{w}) = c$ for each positive constant c, it would look like a circle. Something similar happens if $\boldsymbol{X}^\top \boldsymbol{X}$ is a multiple of the identity matrix. See Figure 0.3(Left) for an illustration.*

Example 5 *Next, consider the case where $\boldsymbol{X}^\top \boldsymbol{X}$ is a diagonal matrix $\boldsymbol{\Lambda} = \mathsf{diag}(\lambda_1, \ldots, \lambda_d)$, so that $\mathcal{Q}(\boldsymbol{w}) = (\boldsymbol{w} - \widehat{\boldsymbol{w}})^\top \boldsymbol{\Lambda}(\boldsymbol{w} - \widehat{\boldsymbol{w}})$. Note that all scalars λ_i are nonnegative, since we have $\lambda_i = \langle X_i, X_i\rangle$. In this case, we may simplify the function of interest as $\mathcal{Q}(\boldsymbol{w}) = \sum_{i=1}^d \lambda_i (w_i - \widehat{w}_i)^2$, and a contour of interest is given by $\sum_{i=1}^d \lambda_i (w_i - \widehat{w}_i)^2 = c$. Think about what this represents in 2-dimensions. When $\lambda_1 < \lambda_2$, it is an ellipse whose major and minor axes coincide with the x and y axes, respectively. The scalar λ_i represents the curvature along axis i: a larger value of λ_i indicates larger curvature along direction i (and a correspondingly narrower contour in that direction). See Figures 0.3 and 0.4 for illustrations.*

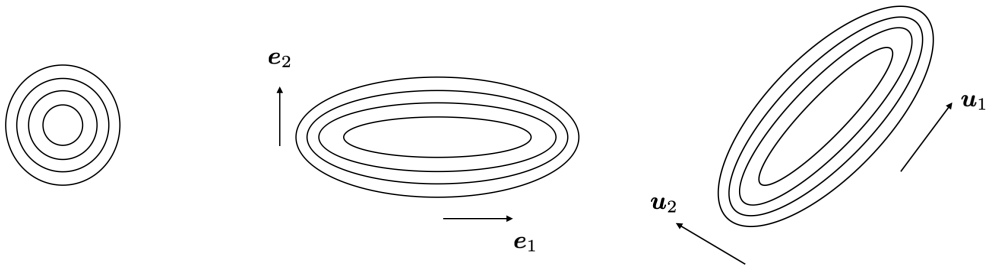

Figure 0.3: Three types of contour plots that parallel the three examples; in all cases, the contours are centered at the point $\widehat{\boldsymbol{w}}$. (Left): This coresponds to the case where the matrix $\boldsymbol{X}^\top \boldsymbol{X} = \boldsymbol{I}$ (or more generally, a multiple of the identity). (Center) This is the case where $\boldsymbol{X}^\top \boldsymbol{X}$ is a diagonal matrix with distinct entries. (Right) The general case, in which we have a basis transformation that makes $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ the major and minor axes, respectively. The contours are otherwise unchanged from the center plot.

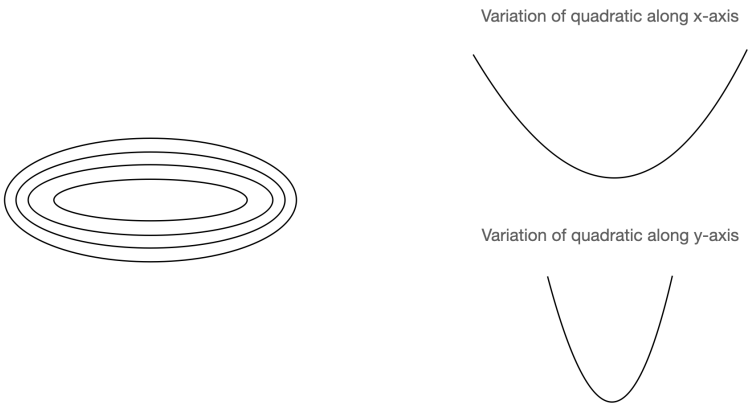

Figure 0.4: Interpreting a contour plot in terms of variation of the function along all the directions. The contour plot on the left has large curvature along the y-axis and lower curvature along the x-axis. The contour is therefore narrower along the y-axis.

Before presenting our final example, we recall a fundamental fact from linear algebra: the eigenvalue decomposition. The eigenvalue decomposition states that any $d \times d$ symmetric matrix $\boldsymbol{M}$ can be decomposed as $\boldsymbol{M} = \boldsymbol{U}^\top \boldsymbol{\Lambda} \boldsymbol{U}$, where $\boldsymbol{U} \in \mathbb{R}^{d \times d}$ is an *orthonormal matrix* satisfying $\boldsymbol{U}\boldsymbol{U}^\top = \boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{I}$. Letting $\boldsymbol{u}_i$ denote the i-th row of $\boldsymbol{U}$ (once again viewed as a $d \times 1$ column vector), the matrix $\boldsymbol{\Lambda}$ is a diagonal matrix containing the *eigenvalues* of $\boldsymbol{M}$. In particular, we have $\boldsymbol{M}\boldsymbol{u}_i = \lambda_i \boldsymbol{u}_i$ for each $1 \leq i \leq d$. This decomposition is unique provided all the eigenvalues are distinct.

Example 6 *Now consider the general case where $\boldsymbol{X}^\top \boldsymbol{X}$ is some $d \times d$, symmetric matrix. Note from HW0 that $\boldsymbol{X}^\top \boldsymbol{X}$ is also positive definite, which is another way of saying that all its eigenvalues are positive. Then we have the eigendecomposition $\boldsymbol{X}^\top \boldsymbol{X} = \boldsymbol{U}^\top \boldsymbol{\Lambda} \boldsymbol{U}$, where the matrix $\boldsymbol{\Lambda}$ is diagonal and contains only positive values. With this decomposition in hand, note that we may write*

$$\begin{aligned}
\mathcal{Q}(\boldsymbol{w}) &= (\boldsymbol{w} - \widehat{\boldsymbol{w}})^\top \boldsymbol{X}^\top \boldsymbol{X} (\boldsymbol{w} - \widehat{\boldsymbol{w}}) \\
&= (\boldsymbol{w} - \widehat{\boldsymbol{w}})^\top \boldsymbol{U}^\top \boldsymbol{\Lambda} \boldsymbol{U} (\boldsymbol{w} - \widehat{\boldsymbol{w}}) \\
&= (\boldsymbol{U}(\boldsymbol{w} - \widehat{\boldsymbol{w}}))^\top \boldsymbol{\Lambda} (\boldsymbol{U}(\boldsymbol{w} - \widehat{\boldsymbol{w}})) \\
&= (\boldsymbol{z} - \widehat{\boldsymbol{z}})^\top \boldsymbol{\Lambda} (\boldsymbol{z} - \widehat{\boldsymbol{z}}),
\end{aligned}$$

where in the last line we have defined $\boldsymbol{z} = \boldsymbol{U}\boldsymbol{w}$ and $\widehat{\boldsymbol{z}} = \boldsymbol{U}\widehat{\boldsymbol{w}}$. Note that the last line looks exactly like the previous example, the level sets will look like ellipses in $\boldsymbol{z}$ space.

Operationally, we just did a basis transformation*. We went from $\boldsymbol{w}$ variables to $\boldsymbol{z}$ variables by "rotating" the coordinate system with the matrix $\boldsymbol{U}$. In this new coordinate system (whose axes are now indexed now by $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_d$ instead of the standard basis vectors $\boldsymbol{e}_1, \ldots, \boldsymbol{e}_d$), our contour plots will look once again like ellipses. See the third panel of Figure 0.3*

In these three examples, we saw how one might visualize a quadratic loss function of several variables. The key property that resulted in a unique minimizer was the positive semidefiniteness of $\boldsymbol{X}^\top \boldsymbol{X}$. More generally, we will see that a (smooth enough) *convex* function f is one that has a positive-semidefinite *Hessian* matrix, where the i, j-th entry of the Hessian matrix is given by

$$[\nabla^2 f]_{i,j} = \frac{\partial f}{\partial w_i \partial w_j}. \tag{0.6}$$

Exercise 2 *Verify that the Hessian matrix of the functions $\mathcal{L}$ and $\mathcal{Q}$ are given by $2\boldsymbol{X}^\top \boldsymbol{X}$.*

0.5. Takeaways

- When does a system of linear equations have a (unique) solution?
- How to set up the linear regression problem under the squared loss.
- The conceptual tools of *lifting* to add on several features to $\boldsymbol{x}$ and *padding ones* to model the intercept. Both of these operations still keep the fitting problem one of least squares.

- Both analytic and geometric interpretations of solving a least squares problem.
- How to visualize the squared loss via contour plots by recalling the eigenvalue decomposition.