

EE 127/227A Study Guide / Tips and Tricks

Joshua Achiam

December 3, 2017

1 Vector Concepts

1.1 Linear Independence

A set of vectors $\{u_1, \dots, u_n\}$, with $u_i \in \mathbb{R}^m$ for all i , is **linearly independent** if and only if no u_i can be expressed as a linear combination of the other vectors in the set.

One way to check for linear independences is as follows: consider the sum

$$y = \sum_{i=1}^n \alpha_i u_i, \quad \forall i, \alpha_i \in \mathbb{R}.$$

First, note that y is a vector in \mathbb{R}^m . If we can pick any coefficients $\alpha_i \neq 0$ and obtain $y = 0$, then the set is **not linearly independent**. When vectors are not linearly independent, we call them **linearly dependent**. To state that symbolically:

$$\exists \{\alpha_1, \dots, \alpha_n\} \text{ not all equal to } 0 : \sum_{i=1}^n \alpha_i u_i = 0 \iff \text{the vectors } u_1, \dots, u_n \text{ are linearly dependent.}$$

Example 1. Consider the vectors

$$u_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad u_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad u_3 = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}.$$

The set of vectors $\{u_1, u_2, u_3\}$ are **linearly dependent** because we can write

$$u_3 = u_1 + u_2;$$

that is, we can write $0 = u_1 + u_2 - u_3$.

The set of vectors $\{u_1, u_2\}$ is **linearly independent** because the only way to get $0 = \alpha_1 u_1 + \alpha_2 u_2$ is if we pick $\alpha_1 = \alpha_2 = 0$.

1.2 Orthogonality and Orthonormality

Two vectors x, y are said to be **orthogonal** if their inner product is zero: $x^T y = 0$.

A set of vectors $\{u_1, \dots, u_n\}$ is said to be orthogonal if $\forall i \neq j, u_i^T u_j = 0$.

A set of vectors $\{u_1, \dots, u_n\}$ is said to be **orthonormal** if it is orthogonal, and the vectors all have unit magnitude—that is, if

$$u_i^T u_j = \begin{cases} 1 & i = j, \\ 0 & i \neq j. \end{cases}$$

1.3 Span

The **span** of a set of vectors $\{u_1, \dots, u_n\}$ with $u_i \in \mathbb{R}^m$ for all i is the set of all the linear combinations of those vectors. Specifically,

$$\text{span}(\{u_1, \dots, u_n\}) = \left\{ \sum_{i=1}^n \alpha_i u_i : \forall i, \alpha_i \in \mathbb{R} \right\}.$$

2 Matrix Concepts

2.1 Dyads

A **dyad** is a matrix equal to the outer product of two vectors. The matrix

$$A = uv^T \quad u \in \mathbb{R}^m, v \in \mathbb{R}^n$$

is a dyad. It has dimensions $A \in \mathbb{R}^{m \times n}$, and components $A_{ij} = u_i v_j$.

When we matrix multiply by a vector $x \in \mathbb{R}^n$, we get the following:

$$Ax = uv^T x = (v^T x)u.$$

To prove that statement, consider the following derivation for the i th component of the vector Ax :

$$\begin{aligned} (Ax)_i &= \sum_{j=1}^n A_{ij} x_j \\ &= \sum_{j=1}^n u_i v_j x_j \\ &= u_i \sum_{j=1}^n v_j x_j \\ \therefore (Ax)_i &= u_i (v^T x). \end{aligned}$$

2.2 Orthogonal Matrix

A matrix is said to be an **orthogonal matrix** if it is square, real-valued, and **its columns are orthonormal**. If U is an orthogonal matrix, then $U^T U = U U^T = I$, where I is the identity matrix.

2.3 Trace

The trace of a square matrix $A \in \mathbb{R}^{n \times n}$ is the sum of its diagonal elements,

$$\text{trace}(A) = \sum_{i=1}^n A_{ii}.$$

The trace of a square matrix is also equal to the sum of its eigenvalues,

$$\text{trace}(A) = \sum_{i=1}^n \lambda_i,$$

where $\lambda_1, \dots, \lambda_n$ are the eigenvalues of A .

2.4 Invertibility

A square matrix is invertible if and only if any of the following equivalent conditions are met:

- it is full rank,
- all of its eigenvalues are nonzero,
- its determinant is nonzero,
- the columns of the matrix are linearly independent.

To be clear: if *any* of these is true, *all* of them are true.

3 Range and Nullspace

3.1 Range

The **range** of a matrix $A \in \mathbb{R}^{m \times n}$ is the set $\mathcal{R}(A) = \{Ax \mid x \in \mathbb{R}^n\}$. The **rank** of a matrix is equal to the dimension of its range: $\text{rank}(A) = \dim \mathcal{R}(A)$.

The range can equivalently be defined as the **span of the columns** of A , also called the column space of A . To see this, let a_1, \dots, a_n be the column vectors of A . That is, write

$$A = \begin{bmatrix} | & & | \\ a_1 & \cdots & a_n \\ | & & | \end{bmatrix}.$$

Then, for all $x \in \mathbb{R}^n$, you have

$$Ax = \begin{bmatrix} | & & | \\ a_1 & \cdots & a_n \\ | & & | \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \sum_{i=1}^n x_i a_i.$$

Thus, $\mathcal{R}(A) = \{Ax \mid x \in \mathbb{R}^n\} = \{\sum_{i=1}^n x_i a_i \mid \forall i, x_i \in \mathbb{R}\} = \text{span}(\{a_1, \dots, a_n\})$.

3.2 Nullspace

The **null space** of a matrix $A \in \mathbb{R}^{m \times n}$ is the set $\mathcal{N}(A) = \{x \mid x \in \mathbb{R}^n, Ax = 0\}$. The **nullity** of a matrix is equal to the dimension of its nullspace: $\text{nullity}(A) = \dim \mathcal{N}(A)$.

3.3 Rank-Nullity Theorem

Theorem 1 (Rank-Nullity Theorem). *For any matrix $A \in \mathbb{R}^{m \times n}$, the sum of the rank and nullity is equal to n . That is,*

$$\text{rank}(A) + \text{nullity}(A) = \dim \mathcal{R}(A) + \dim \mathcal{N}(A) = n.$$

3.4 Examples

Example 2 (Simple Ranges and Null Spaces). Consider the following simple matrices:

$$(a) \ A = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}, \quad (b) \ A = \begin{bmatrix} \lambda_1 & 0 \\ 0 & 0 \end{bmatrix}, \quad (c) \ A = uv^T, \ u \in \mathbb{R}^m, v \in \mathbb{R}^n,$$

where $\lambda_1, \lambda_2 \neq 0$, and $u, v \neq 0$. What are the ranges and nullspaces of these matrices?

First, consider case (a). Here, every vector in \mathbb{R}^2 is in the range space! Direct proof: let y be any vector in \mathbb{R}^2 . Then for $x = [\lambda_1^{-1}y_1, \lambda_2^{-1}y_2]^T$, we have $y = Ax$, so y is in the range space. Thus, $\mathcal{R}(A) = \mathbb{R}^2$.

As for the nullspace: for any x , we have $y = Ax = [\lambda_1 x_1, \lambda_2 x_2]^T$. Because $\lambda_1, \lambda_2 \neq 0$, we can only get $y = 0$ if both x_1 and x_2 are equal to 0. Thus, $\mathcal{N}(A) = \{0\}$.

Consider case (b). We can immediately see that any vector proportional to $e_2 = [0, 1]^T$ is in the null space. Proof: let $v = \alpha e_2$, where $\alpha \in \mathbb{R}$ is any scalar coefficient. Then $Av = 0$. Thus, $\mathcal{N}(A) = \text{span}(\{e_2\})$. As for the range of A : let $x = [x_1, x_2]^T$. Observe that $Ax = [\lambda_1 x_1, 0]^T$. So for any $y = [y_1, 0]$, we can pick $x = [\lambda_1^{-1}y_1, 0]$, and then $y = Ax$, so $y \in \mathcal{R}(A)$. From this, we find that $\mathcal{R}(A) = \text{span}(\{e_1\})$, where $e_1 = [1, 0]$.

For case (c), we can compute the range space very easily:

$$\begin{aligned} \mathcal{R}(A) &= \{Ax \mid x \in \mathbb{R}^n\} \\ &= \{uv^T x \mid x \in \mathbb{R}^n\} \\ &= \{(v^T x)u \mid x \in \mathbb{R}^n\} \\ &=^\dagger \text{span}(\{u\}). \end{aligned}$$

[†] To see that last step, let $y = \alpha u$ be a vector in the span of $\{u\}$. For $x = \alpha v / \|v\|_2^2$, we have $Ax = uv^T(\alpha v / \|v\|_2^2) = \alpha u$. Thus, $\text{span}(\{u\}) \subseteq \mathcal{R}(A)$. From the third step, we see that every element in $\mathcal{R}(A)$ is in the span of A , because it has the form $(v^T x)u$ for some x , thus $\mathcal{R}(A) \subseteq \text{span}(\{u\})$. This proves the equality.

The nullspace for (c) is the set of all vectors orthogonal to v : $\mathcal{N}(A) = \{x \in \mathbb{R}^n \mid v^T x = 0\}$. You can see this by noting that if $v^T x = 0$, then $Ax = uv^T x = (v^T x)u = 0$.

4 Fundamental Theorem of Linear Algebra

Theorem 2 (Fundamental Theorem of Linear Algebra). *For any matrix $A \in \mathbb{R}^{m \times n}$,*

$$\mathcal{N}(A) \overset{\perp}{\oplus} \mathcal{R}(A^T) = \mathbb{R}^n.$$

What does this theorem tell us? What does it mean? To explain this, we have to first define the notation in the problem. The \oplus symbol means **direct sum**. Let $X, Y \subseteq \mathbb{R}^n$ be two vector subspaces. The space Z is said to be the direct sum of X and Y if and only if every vector in $z \in Z$ can be written as a sum $z = x + y$, with $x \in X, y \in Y$, and the choice of x and y is *unique*; also, every vector $x + y$ must be in Z . So, when the uniqueness conditions are satisfied,

$$X \oplus Y = \{x + y | x \in X, y \in Y\}.$$

The notation $\overset{\perp}{\oplus}$ means that the two subspaces we are direct-summing are *orthogonal*: $\mathcal{N}(A) \perp \mathcal{R}(A^T)$. This means that $\forall x \in \mathcal{N}(A), \forall y \in \mathcal{R}(A^T), x \perp y$ (or equivalently, $x^T y = 0$).

4.1 How to Apply Fundamental Theorem of Linear Algebra

You will use this theorem primarily for one of two reasons:

- to get the range of A when you only know the null space of A , or to get the null space when you only know the range,
- or as a proof step, when you have to recognize that $x \in \mathcal{N}(A) \implies x \perp \mathcal{R}(A^T)$, or some equivalent statement.

4.2 Getting the range or nullspace when you only know the other

Correct way: Suppose that $V = \{v_1, \dots, v_r\}$ is an orthonormal basis for $\mathcal{R}(A^T)$ —that is, $\text{span}(V) = \mathcal{R}(A^T)$. Because $\mathcal{R}(A^T) \overset{\perp}{\oplus} \mathcal{N}(A) = \mathbb{R}^n$, any linearly independent vectors $V' = \{v_{r+1}, \dots, v_n\}$ that are all perpendicular to the vectors in V constitute a basis for $\mathcal{N}(A)$, so $\mathcal{N}(A) = \text{span}(V')$.

Incorrect way: $\mathcal{N}(A) = \mathbb{R}^n \setminus \mathcal{R}(A^T)$. This is totally wrong. Do not write this.

Example 3 (Why the incorrect way is incorrect). Consider the matrix

$$A = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}.$$

We can see that $\mathcal{R}(A^T) = \text{span}(\{e_1\})$, where $e_1 = [1, 0]^T$. The nullspace is $\mathcal{N}(A) = \text{span}(\{e_2\})$, where $e_2 = [0, 1]^T$. But what is the set $\mathbb{R}^2 \setminus \mathcal{R}(A^T)$?

$$\begin{aligned} \mathbb{R}^2 \setminus \mathcal{R}(A^T) &= \{x \in \mathbb{R}^2 \mid x \notin \mathcal{R}(A^T)\} \\ &= \{x \in \mathbb{R}^2 \mid x \not\propto e_1\} \end{aligned}$$

That is, $\mathbb{R}^2 \setminus \mathcal{R}(A^T)$ is the entire plane, minus the line along the e_1 -axis. This is *not* equivalent to the nullspace, which is *only* the line along the e_2 -axis.

5 Eigenvalue Decomposition and Singular Value Decomposition

5.1 Eigendecomposition

For real, symmetric matrices $A \in \mathbb{S}^n$, A has an eigendecomposition

$$A = V\Lambda V^T = \sum_{i=1}^n \lambda_i v_i v_i^T,$$

where V is the matrix whose columns are the orthonormal eigenvectors v_i . To recap on matrix sizes:

- $V = [v_1, \dots, v_n] \in \mathbb{R}^{n \times n}$
- $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_n) \in \mathbb{R}^{n \times n}$.

5.2 SVD

5.2.1 Compact Form

When we have a nonsquare matrix, we do not have an eigendecomposition, but we *do* have something similar: the singular value decomposition (SVD). Consider a matrix $A \in \mathbb{R}^{m \times n}$ with rank r . It has an SVD given by

$$A = U\Sigma V^T = \sum_{i=1}^r \sigma_i u_i v_i^T,$$

where $\sigma_1 > \sigma_2 > \dots > \sigma_r > 0$, $\{u_1, \dots, u_r\}$ are orthonormal vectors in \mathbb{R}^m , and $\{v_1, \dots, v_r\}$ are orthonormal vectors in \mathbb{R}^n . On matrix sizes:

- $U = [u_1, \dots, u_r] \in \mathbb{R}^{m \times r}$,
- $\Sigma = \text{diag}(\sigma_1, \dots, \sigma_r) \in \mathbb{R}^{r \times r}$,
- $V = [v_1, \dots, v_r] \in \mathbb{R}^{n \times r}$.

Note that $U^T U = V^T V = I_r$. However, also note that when $m > r$ and $n > r$, $U U^T \neq I_m$, and $V V^T \neq I_n$. (Too see this, recognize that U and V are rank r , so $U U^T$ and $V V^T$ can be at most rank r , but the identity matrix I_m has rank $m > r$, and I_n has rank $n > r$.)

The vectors u_1, \dots, u_r are called the **left singular vectors** of A , and the vectors v_1, \dots, v_r are called the **right singular vectors** of A .

Important note: **singular values must be strictly positive**, unlike eigenvalues, which can be positive, zero, or negative.

5.2.2 Expanded Form

You can also express the SVD as

$$A = \tilde{U} \tilde{\Sigma} \tilde{V}, \quad (1)$$

where

- $\tilde{U} = [U \ U_+] \in \mathbb{R}^{m \times m}$, where $U_+ \in \mathbb{R}^{m \times (m-r)}$ is a matrix chosen so that \tilde{U} is orthogonal (U_+ has columns u_{r+1}, \dots, u_m which complete a basis with u_1, \dots, u_r , so that $\tilde{U} \tilde{U}^T = \tilde{U}^T \tilde{U} = I_m$).
- $\tilde{\Sigma} = \begin{bmatrix} \Sigma & 0_{r \times (n-r)} \\ 0_{(m-r) \times r} & 0_{(m-r) \times (n-r)} \end{bmatrix} \in \mathbb{R}^{m \times n}$,
- $\tilde{V} = [V \ V_+] \in \mathbb{R}^{n \times n}$, where $V_+ \in \mathbb{R}^{n \times (n-r)}$ is a matrix chosen so that \tilde{V} is orthogonal (V_+ has columns v_{r+1}, \dots, v_n which complete a basis with v_1, \dots, v_r , so that $\tilde{V} \tilde{V}^T = \tilde{V}^T \tilde{V} = I_n$).

5.3 Recovering U or V given A , Σ , and the other

Starting from the $A = U \Sigma V^T$ definition above, we can derive some other relationships which would allow us to obtain U or V , if we had the other matrices.

$$\begin{aligned} A V \Sigma^{-1} &= U \Sigma V^T V \Sigma^{-1} \\ &\stackrel{(a)}{=} U \Sigma \Sigma^{-1} \\ \therefore A V \Sigma^{-1} &= U \end{aligned}$$

where (a) obtains from the fact that $V^T V = I_r$. Check the dimensions of these matrix multiplications—why did we have to right-multiply by V ?

Also,

$$\begin{aligned}\Sigma^{-1}U^T A &= \Sigma^{-1}U^T U \Sigma V^T \\ &= \Sigma^{-1} \Sigma V^T \\ \therefore A^T U \Sigma^{-1} &= V.\end{aligned}$$

5.4 Recovering u_i or v_i given A , σ_i , and the other

We can derive vector relationships that are directly analagous to the matrix relationships in the previous subsection.

$$\begin{aligned}\frac{1}{\sigma_i} A v_i &= \frac{1}{\sigma_i} \left(\sum_{j=1}^r \sigma_j u_j v_j^T \right) v_i \\ &= \frac{1}{\sigma_i} \sum_{j=1}^r \sigma_j u_j (v_j^T v_i) \\ &\stackrel{(a)}{=} \frac{1}{\sigma_i} \sum_{j=1}^r \sigma_j u_j \delta_{ij} \\ &= \frac{1}{\sigma_i} \sigma_i u_i \\ \therefore \frac{1}{\sigma_i} A v_i &= u_i.\end{aligned}$$

The step at (a) comes from recalling that the v_i are orthonormal vectors: $v_i^T v_j = 0$ if $i \neq j$, and $v_i^T v_j = 1$ if $i = j$. We summarize this with the symbol δ_{ij} ¹, which is equal to 1 when $i = j$ and 0 otherwise.

By the same strategy, we can derive the relationship

$$\frac{1}{\sigma_i} A^T u_i = v_i.$$

You should completely understand the proof above, and be able to synthesize the proof for the latter statement by yourself. On a test, if you have to recover v_i or u_i given A , σ_i , and the other, you should prove that your answer is correct. The proof strategy here is a valid way to do that.

¹This is called the Kronecker delta.

5.5 Connecting SVD to Eigendecomposition

Consider an arbitrary nonsquare matrix $A \in \mathbb{R}^{m \times n}$ with SVD $A = \sum_{i=1}^r \sigma_i u_i v_i^T$. We will show that the singular values of A and the eigenvalues of $A^T A$ (and AA^T) are connected.

Fact 1. *The eigenvalues of the matrix $A^T A$ (and AA^T) are equal to the squares of the singular values of A . That is, if σ_i is a singular value of A , then $\lambda_i = \sigma_i^2$ is an eigenvalue of $A^T A$ (and AA^T).*

Note that

$$\begin{aligned} A &= \tilde{U} \tilde{\Sigma} \tilde{V}^T \\ \implies A^T A &= \tilde{V} \tilde{\Sigma}^T \tilde{U}^T \tilde{U} \tilde{\Sigma} \tilde{V}^T \\ &= \tilde{V} \tilde{\Sigma}^T \tilde{\Sigma} \tilde{V}^T \\ &= \tilde{V} \begin{bmatrix} \Sigma & 0_{r \times (m-r)} \\ 0_{(n-r) \times r} & 0_{(n-r) \times (m-r)} \end{bmatrix} \begin{bmatrix} \Sigma & 0_{r \times (n-r)} \\ 0_{(m-r) \times r} & 0_{(m-r) \times (n-r)} \end{bmatrix} \tilde{V}^T \\ &= \tilde{V} \begin{bmatrix} \Sigma^2 & 0_{r \times (n-r)} \\ 0_{(n-r) \times r} & 0_{(n-r) \times (n-r)} \end{bmatrix} \tilde{V}^T. \end{aligned}$$

Observe that this specifies an eigendecomposition for $A^T A$, with eigenvalues σ_i^2 ! If you don't like the matrix form, let's do it another way. We derive the following:

$$\begin{aligned} A^T A &= \left(\sum_{i=1}^r \sigma_i u_i v_i^T \right)^T \left(\sum_{j=1}^r \sigma_j u_j v_j^T \right) \\ &= \sum_{i=1}^r \sum_{j=1}^r \sigma_i \sigma_j v_i u_i^T u_j v_j^T \\ &= \sum_{i=1}^r \sum_{j=1}^r \sigma_i \sigma_j v_i (u_i^T u_j) v_j^T \\ &\stackrel{(a)}{=} \sum_{i=1}^r \sum_{j=1}^r \sigma_i \sigma_j v_i \delta_{ij} v_j^T \\ &= \sum_{i=1}^r \sigma_i^2 v_i v_i^T \end{aligned}$$

This also proves the result. For (a), recall that the u_i are orthonormal. The symbol δ_{ij} , called the Kronecker delta, is defined so that $\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ otherwise.

There is another connection worth pointing out.

Fact 2. Consider a square, symmetric matrix $A \in \mathbb{S}^n$. The absolute values of the nonzero eigenvalues of A are its singular values.

To see this, note the following:

$$\begin{aligned} A &= \sum_{i=1}^n \lambda_i v_i v_i^T \\ &= \sum_{i=1}^n |\lambda_i| (\operatorname{sgn}(\lambda_i) v_i) v_i^T. \end{aligned}$$

Define $u_i \doteq \operatorname{sgn}(\lambda_i) v_i$, and choose the ordering so that $|\lambda_1| > |\lambda_2| > \dots$. Choose $\sigma_i = |\lambda_i|$ for $i = 1, \dots, r$ where r is the rank of A (equivalently, r is the index of the last nonzero eigenvalue). This is an SVD for A . (Recall that singular values have to be strictly positive.)

6 The Most Important Equation in Linear Algebra: $Ax = b$

Here, we consider one of the most important questions in linear algebra that you need to know the answer to: given a matrix $A \in \mathbb{R}^{m \times n}$ and a vector $b \in \mathbb{R}^m$, what choices of x solve $Ax = b$?

6.1 Case 1: A is invertible (One Solution)

Suppose $m = n$ and A is full rank, so A is invertible. Then there is a single, unique solution, and it is

$$x = A^{-1}b.$$

6.2 Case 2: $b \notin \mathcal{R}(A)$ (Zero Solutions)

The statement $b \in \mathcal{R}(A)$ means that there is some x such that $b = Ax$. If $b \notin \mathcal{R}(A)$, **then there is no x such that $Ax = b$.**

6.3 Case 3: A is not invertible, but $b \in \mathcal{R}(A)$ (Infinity Solutions)

In this case, there is at least some $x_0 \in \mathbb{R}^n$ such that $Ax_0 = b$ (because $b \in \mathcal{R}(A)$), but A is not invertible, so it is not *unique*.

Recall that A being not invertible means that A has some eigenvalues equal to zero, which means that the nullspace of A is not empty. Thus, for all $v \in \mathcal{N}(A)$, $x_0 + v$ is a solution. To see this, note that

$$\begin{aligned}\forall v \in \mathcal{N}(A), Ax_0 + Av &= Ax_0 + Av \\ &= b + 0 \\ &= b.\end{aligned}$$

How do we find an x_0 that satisfies the equation $Ax = b$ in the first place, though? We can use the SVD to accomplish this. If $A = U\Sigma V^T$ is an SVD of A , then a pseudo-inverse of A can be obtained as $A^\dagger = V\Sigma^{-1}U^T$. Picking $x_0 = A^\dagger b$ gives us a solution.

7 Projections

In the **projection problem**, you are given some set of vectors \mathcal{X} , and some vector v , and asked to find the projection of v onto \mathcal{X} . Concretely, you are being asked to solve the problem

$$x^* = \arg \min_{x \in \mathcal{X}} \|v - x\|.$$

That is, you want to find the element x^* in \mathcal{X} which is as close as possible to v in some sense.

7.1 Projecting onto a line

A line is a set $\mathcal{L}(x_0, u) = \{x : x = x_0 + tu, t \in \mathbb{R}\}$. Here, x_0 is just some point that lies on the line, and u is a direction (which we will take to have unit norm, without loss of generality, though this is not necessary).

In the problem of **projecting onto a line**, we have some point $v \in \mathbb{R}^n$, and we would like to find the closest point to v on the line, $x^* \in \mathcal{L}(x_0, u)$, in the sense of Euclidean distance (L_2 -norm). We first set up the optimization problem:

$$x^* = \arg \min_{x \in \mathcal{L}(x_0, u)} \|v - x\|_2.$$

Now, we write out the constraint explicitly—this involves adding a variable t :

$$\begin{aligned} x^* &= \arg \min_{x, t} \|v - x\|_2. \\ \text{s.t. } &x = x_0 + tu \end{aligned}$$

Observe that if we substitute $x = x_0 + tu$ in the objective, we can remove the variable x from the optimization problem, and solve a new problem that is entirely in t . The optimal x relates to the optimal t according to

$$x^* = x_0 + t^*u,$$

and the optimal t is the solution to

$$t^* = \arg \min_t \|v - (x_0 + tu)\|_2.$$

This last optimization problem is easy to solve. First, square the objective (which in this case does not change the solution—can you see why?), and then expand it:

$$\begin{aligned} t^* &= \arg \min_t \|v - (x_0 + tu)\|_2^2 \\ &= \arg \min_t (v - x_0 - tu)^T (v - x_0 - tu) \\ &= \arg \min_t \|v - x_0\|_2^2 + t^2 \|u\|_2^2 - 2tu^T(v - x_0). \end{aligned}$$

The optimization problem is now a quadratic, convex program with a single scalar variable. Setting the gradient equal to zero and solving for t^* obtains the solution:

$$t^* = \frac{u^T(v - x_0)}{\|u\|_2^2}.$$

When u has unit norm, this simplifies to $t^* = u^T(v - x_0)$.

7.2 Projecting onto a plane

TODO

7.3 Projection onto a subspace

TODO

7.4 Vectors and Tensors: Invariance Under Rotations

TODO

8 Optimization Concepts

8.1 You Can Optimize In Any Order

Consider the following optimization problem:

$$\min_{x \in \mathcal{X}, y \in \mathcal{Y}} f(x, y).$$

Define $h(x) \doteq \min_{y \in \mathcal{Y}} f(x, y)$ and $g(y) \doteq \min_{x \in \mathcal{X}} f(x, y)$. Then:

$$\min_{x \in \mathcal{X}, y \in \mathcal{Y}} f(x, y) = \min_{x \in \mathcal{X}} h(x) = \min_{y \in \mathcal{Y}} g(y);$$

that is, you can optimize over x and y in either order and obtain the same result.

Observe that solving $\min_{y \in \mathcal{Y}} f(x, y)$ gives you an optimal point y^* that depends on the choice of x . We denote this as $y^*(x)$. Then $h(x) = f(x, y^*(x))$.

Similarly, solving $\min_{x \in \mathcal{X}} f(x, y)$ gives you an optimal point x^* that depends on the choice of y . We denote this as $x^*(y)$. Then $g(y) = f(x^*(y), y)$.

8.1.1 What about constraints?

Consider the constrained case, where x and y are related by some constraint—for example, $q(x, y) = 0$. The way to handle this is to fold the constraint into the inner optimization problem, and assign the inner optimization problem a value of $+\infty$ when it is infeasible (if it is a minimization—otherwise $-\infty$). That is, we consider the problem

$$\min_{x \in \mathcal{X}, y \in \mathcal{Y}} f(x, y) \quad : \quad q(x, y) = 0.$$

We define the following:

$$h(x) \doteq \begin{cases} \min_{y \in \mathcal{Y}} f(x, y) & : \quad q(x, y) = 0 \\ +\infty & \text{otherwise.} \end{cases} \quad \begin{matrix} \exists y \in \mathcal{Y} : q(x, y) = 0, \\ \text{otherwise.} \end{matrix}$$

$$g(y) \doteq \begin{cases} \min_{x \in \mathcal{X}} f(x, y) & : \quad q(x, y) = 0 \\ +\infty & \text{otherwise.} \end{cases} \quad \begin{matrix} \exists x \in \mathcal{X} : q(x, y) = 0, \\ \text{otherwise.} \end{matrix}$$

Then as before,

$$\min_{x \in \mathcal{X}, y \in \mathcal{Y}} f(x, y) = \min_{x \in \mathcal{X}} h(x) = \min_{y \in \mathcal{Y}} g(y).$$

8.1.2 What if I have a min and a max? Does this still work?

Consider the optimization problem

$$\min_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} f(x, y).$$

A reasonable question to ask: if you switch the order of optimization here, does it make a difference? The answer is: **unless special conditions are met, you cannot switch the orders of min and max**. Later in the course, we will expand on this—it is a topic that deserves special attention.

8.2 Solving Simple Unconstrained Optimization Problems

When you are given a problem of the form

$$\min_{x \in \mathbb{R}^n} f(x),$$

where there are no constraints on x , and f is fully differentiable everywhere, and f has an important property called **convexity**, a sufficient condition for the point x^* to be an optimal point is

$$\nabla_x f(x^*) = 0.$$

When you have obtained an optimal point x^* , the optimal value is computed to be $f(x^*)$.

Later in the course, we will go into more depth on convexity, but for now, think of it like this: **a convex function has only one local minimum, which is its global minimum, and no other minima**. An upward-facing bowl is convex. A downward-facing bowl is not (it is concave).

8.3 Example: Single-Variable Quadratic Optimization

Consider the problem

$$\min_{x \in \mathbb{R}} f(x) = ax^2 + bx + c,$$

with $a > 0$, so that the parabola is upward-facing, and there is a unique global minimizer.

We can find the minimum by setting the derivative of f to zero and solving:

$$\left. \frac{df}{dx} \right|_{x^*} = 2ax^* + b = 0 \quad \implies \quad x^* = \frac{-b}{2a}.$$

8.4 Example: Multivariable Least-Squares

In the multivariable least-squares problem, we are trying to solve the optimization

$$x^* = \arg \min_{x \in \mathbb{R}^n} \|Ax - b\|_2^2,$$

where $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$. For now, take it for granted that **the objective function is convex**.

To solve this, we first rewrite the objective function:

$$\begin{aligned} \|Ax - b\|_2^2 &= (Ax - b)^T (Ax - b) \\ &= x^T A^T A x - 2b^T A x + b^T b. \end{aligned}$$

Next, we examine the sufficient condition $\nabla_x f(x^*) = 0$. We compute the gradient to be:

$$\nabla_x f(x) = 2A^T A x - 2A^T b.$$

Any solution to the equation $A^T A x = A^T b$ is an optimal point.

8.5 Example: L_2 -Regularized Least-Squares

In the L_2 -regularized version of the multivariable least-squares problem, we are trying to solve the optimization

$$x^* = \arg \min_{x \in \mathbb{R}^n} \|Ax - b\|_2^2 + \lambda \|x\|_2^2,$$

where $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, and $\lambda > 0$.

To solve this, we first rewrite the objective function:

$$\begin{aligned} \|Ax - b\|_2^2 + \lambda \|x\|_2^2 &= (Ax - b)^T (Ax - b) + \lambda x^T x \\ &= x^T A^T A x - 2b^T A x + b^T b + x^T (\lambda I) x \\ &= x^T (A^T A + \lambda I) x - 2b^T A x + b^T b. \end{aligned}$$

We proceed in the same way as in the unregularized case, and determine that any x which satisfies

$$\nabla_x f(x) = 2(A^T A + \lambda I)x - 2A^T b = 0$$

is an optimal point.

9 Convex Sets

Required reading: Boyd Ch. 2, p 22-38

A set of points, $\mathcal{X} \subset \mathbb{R}^n$, is **convex** if for any two points $x_1, x_2 \in \mathcal{X}$, every point on the line segment between x_1 and x_2 is also in \mathcal{X} . In symbols, this condition is:

$$\forall x_1, x_2 \in \mathcal{X}, \forall \alpha \in [0, 1], \quad \alpha x_1 + (1 - \alpha)x_2 \in \mathcal{X}.$$

9.1 Examples of Convex Sets

9.1.1 Affine Sets

A set \mathcal{X} is **affine** if for any two points $x_1, x_2 \in \mathcal{X}$, every point *on the entire line* containing x_1 and x_2 is also in \mathcal{X} . Mathematically:

$$\forall x_1, x_2 \in \mathcal{X}, \forall \alpha \in \mathbb{R}, \quad \alpha x_1 + (1 - \alpha)x_2 \in \mathcal{X}.$$

Clearly, **every affine set is convex**. However, **not every convex set is affine**.

Example 4. Consider the set of solutions to a system of linear equations: $\mathcal{X} \doteq \{x : Ax = b\}$. Is this set affine? Suppose that $x_1, x_2 \in \mathcal{X}$. Thus, we have $Ax_1 = b$ and $Ax_2 = b$. Consider any point $x(\alpha) \doteq \alpha x_1 + (1 - \alpha)x_2$. We compute

$$\begin{aligned} Ax(\alpha) &= A(\alpha x_1 + (1 - \alpha)x_2) \\ &= \alpha(Ax_1) + (1 - \alpha)(Ax_2) \\ &= \alpha b + (1 - \alpha)b \\ &= b. \end{aligned}$$

Thus, $x(\alpha) \in \mathcal{X}$, and \mathcal{X} is affine.

9.1.2 Hyperplanes and Halfspaces

A **hyperplane** is a set of the form

$$\{x : a^T x = b\},$$

for some fixed vector a and fixed scalar b . **Hyperplanes are convex sets**. To prove this, observe that a hyperplane is an affine set by definition.

A **halfspace** is a set of the form

$$\{x : a^T x \leq b\}.$$

Geometrically speaking, a halfspace is a hyperplane and all of the points to one side of it. A hyperplane is also a convex set, and the proof is similar. Let x_1, x_2 be points in a halfspace; for any $\alpha \in [0, 1]$, we have

$$\begin{aligned} a^T(\alpha x_1 + (1 - \alpha)x_2) &= \alpha a^T x_1 + (1 - \alpha)a^T x_2 \\ &\leq \alpha b + (1 - \alpha)b \\ &= b. \end{aligned}$$

Thus every point on the line from x_1 to x_2 is in the halfspace, and the halfspace is convex.

9.1.3 Euclidean Balls

A Euclidean ball centered on x_c with radius r , $B(x_c, r)$, is defined to be the set of points

$$B(x_c, r) \doteq \{x_c + ur : \|u\|_2 \leq 1\}.$$

This set is convex. The easy proof is geometric—just look at the ball, it is clearly convex—but we can also do it the hard way. Consider any points $x_1 = x_c + u_1 r$, $x_2 = x_c + u_2 r$, and any $\alpha \in [0, 1]$. Consider the point $x(\alpha) \doteq \alpha x_1 + (1 - \alpha)x_2$. We have:

$$\begin{aligned} x(\alpha) &= \alpha x_1 + (1 - \alpha)x_2 \\ &= \alpha(x_c + u_1 r) + (1 - \alpha)(x_c + u_2 r) \\ &= x_c + (\alpha u_1 + (1 - \alpha)u_2) r. \end{aligned}$$

Thus $x(\alpha)$ is in the ball if $u(\alpha) \doteq \alpha u_1 + (1 - \alpha)u_2$ satisfies $\|u(\alpha)\|_2 \leq 1$. To check this, we proceed:

$$\begin{aligned} \|u(\alpha)\|_2^2 &= \alpha^2 \|u_1\|_2^2 + (1 - \alpha)^2 \|u_2\|_2^2 + 2\alpha(1 - \alpha)u_1^T u_2 \\ &\stackrel{(a)}{\leq} \alpha^2 + (1 - \alpha)^2 + 2\alpha(1 - \alpha) \\ &= \alpha^2 + 1 + \alpha^2 - 2\alpha + 2\alpha - 2\alpha^2 \\ &= 1. \end{aligned}$$

The step (a) makes use of two facts: 1) by definition, $\|u_1\|_2^2 \leq 1$ and $\|u_2\|_2^2 \leq 1$; and 2) the Cauchy-Schwarz inequality gives $u_1^T u_2 \leq \|u_1\|_2 \|u_2\|_2$. The last step is just algebra. We conclude that $x(\alpha) \in B(x_c, r)$, and thus that $B(x_c, r)$ is a convex set.

Remark. We can also equivalently define the ball by the expression

$$B(x_c, r) \doteq \{x : \|x - x_c\|_2 \leq r\}.$$

9.2 Operations Preserving Convexity

Proposition 1. *Let $\mathcal{X}_1, \mathcal{X}_2$ be two convex sets. The intersection, $\mathcal{X} \doteq \mathcal{X}_1 \cap \mathcal{X}_2$, is also convex.*

Proof. Consider points $x_1, x_2 \in \mathcal{X} = \mathcal{X}_1 \cap \mathcal{X}_2$. Because x_1 and x_2 are in the intersection of $\mathcal{X}_1, \mathcal{X}_2$, both points must be in both sets. Because the sets are convex, we have, for all $\alpha \in [0, 1]$,

$$\begin{aligned}\alpha x_1 + (1 - \alpha)x_2 &\in \mathcal{X}_1, \\ \alpha x_1 + (1 - \alpha)x_2 &\in \mathcal{X}_2.\end{aligned}$$

It follows immediately that $\alpha x_1 + (1 - \alpha)x_2$ is in \mathcal{X} . Therefore \mathcal{X} satisfies the convexity definition and is convex. \square

Example 5 (Polyhedron). A **polyhedron** is an intersection of halfspaces. Consider k halfspaces $\mathcal{H}_i \doteq \{x : a_i^T x \leq b_i\}$, for $i = 1, \dots, k$. The polyhedron

$$\mathcal{P} \doteq \bigcap_{i=1}^k \mathcal{H}_i$$

is a convex set.

Proposition 2. *Let \mathcal{X} be a convex set, and let f be an affine function (it has the form $f(x) = Ax + b$ for some A, b). The image of \mathcal{X} under f ,*

$$f(\mathcal{X}) \doteq \{f(x) : x \in \mathcal{X}\},$$

is convex.

Proof is left as an exercise to the reader. (It's really easy though!)

Example 6 (Ellipsoid). One way to define an ellipsoid is by the set

$$\mathcal{E}(x_c, A) \doteq \{x_c + Au \mid \|u\|_2 \leq 1\}.$$

This is the image of the unit ball $B(x_c, 1)$ under the affine function $f(x) = Ax + (I - A)x_c$. To see this, note that $f(x_c + u) = A(x_c + u) + (I - A)x_c = x_c + Au$. Because $B(x_c, 1)$ is a convex set, and $\mathcal{E}(x_c, A)$ is the image of $B(x_c, 1)$ under an affine function, $\mathcal{E}(x_c, A)$ is also convex.

10 Convex Functions

Required reading: Boyd Ch. 3, p 67-87.

A function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is a **convex function** if $\text{dom } f$ is a convex set, and for all points $x, y \in \text{dom } f$, and all $\alpha \in [0, 1]$, we have

$$f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y).$$

A function is **concave** if $-f$ is convex.

10.1 First-Order Conditions for Convexity

Suppose f is differentiable. Then f is convex if and only if $\text{dom } f$ is convex, and for every $x, y \in \text{dom } f$,

$$f(y) \geq f(x) + \nabla f(x)^T(y - x).$$

This is saying that the linearization of f at x is a *global underestimator* for f .

10.2 Second-Order Conditions for Convexity

Suppose f is twice-differentiable. Then f is convex if and only if $\text{dom } f$ is convex and the Hessian of f , $\nabla^2 f$, is positive semi-definite everywhere: $\forall x \in \text{dom } f$,

$$\nabla^2 f(x) \succeq 0.$$

10.3 Examples of Convex Functions

The following are some examples of convex functions of scalar variables.

- *Exponential.* $f(x) = e^{ax}$ is convex on \mathbb{R} , for any $a \in \mathbb{R}$.
- *Powers.* $f(x) = x^a$ is convex on \mathbb{R}_{++} , for any $a \geq 1$ or $a \leq 0$. Note: this is not convex on the entire real line, just on \mathbb{R}_{++} . Think about x^3 —is this convex on the set $\{x \leq 0\}$? (It is not. Check first-order conditions!)

- *Logarithm.* $f(x) = -\log x$ is convex on \mathbb{R}_{++}

Some examples of convex vector-valued functions

- *Norms.* All norms on \mathbb{R}^n are convex (note this follows from the triangle inequality and using it to directly prove $f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y)$).
- *Max function.* $f(x) = \max\{x_1, \dots, x_n\}$ is convex on \mathbb{R}^n .
- *Random function.* $f(x, y) = x^2/y$ is convex on $\mathbb{R} \times \mathbb{R}_{++}$ (show that the hessian is PSD).
- *Affine function.* $f(x) = a^\top x + b$ is convex on \mathbb{R}^n .

10.4 Operations that preserve convexity

Usually in optimization, we have functions that are composed with one another and not nice functions that we can directly compute Hessians or use the definition of convexity to show functions are convex. We revert to rules that preserve convexity. Below are some key operations that preserve convexity. All the rules below are worth memorizing.

- Nonnegative weighted sum of convex functions: $\sum_{i=1}^n a_i f_i$ $a_i \geq 0$ and f_i convex $\forall i$
(extends to infinite sums, integrals)
- Composition with affine function: f convex $\Rightarrow f(Ax + b)$ is convex
- Pointwise maximum of convex functions: if f_i are convex for $i = 1, \dots, n$ then $f(x) = \max\{f_1(x), \dots, f_m(x)\}$ is convex
- Pointwise supremum: if $f(x, y)$ is convex in x , then for $y \in C$ $g(x) = \sup_{y \in C} f(x, y)$ is convex (note C does not need to be convex!) **This rule you will use repeatedly**
- Pointwise Infimum: if $f(x, y)$ is jointly-convex in (x, y) (not the same as separately being convex in x and in y) and C is a convex set then $g(x) = \inf_{y \in C} f(x, y)$ is convex
- Composition: g is convex, h convex and nondecreasing, then $f(x) = h(g(x))$ is convex
- Composition: g is concave, h convex and nonincreasing, then $f(x) = h(g(x))$ is convex

- Composition: g is concave, h is concave and nondecreasing, then $f(x) = h(g(x))$ is concave
- Composition: g is convex, h is concave and nonincreasing, then $f(x) = h(g(x))$ is concave.

Note for the compositions, we need to be careful about the range of $g(x)$ and domain of $h(x)$ - if the range of $g(x)$ contains values that are not in the domain of $h(x)$ then $h(g(x))$ is undefined.

10.5 Examples of proving convexity using operations that preserve convexity

- **Least squares objective:** Consider $f(x) = \|Ax - b\|_2^2$. This function is **convex**. Recall that the L_2 norm is a convex function, and composition with an affine function preserves convexity. Observe that f is just the composition of L_2 norm with an affine function.
- **A certain eigenvalue function:** Consider $f(x) = \lambda_{\min}(A(x))$, where $A(x) = \sum_{i=1}^n x_i A_i$, and each $A_i \in \mathbb{S}^n$. This is a fairly strange-looking function, to say the least, and it is not immediately clear how to analyze it from first or second-order conditions. Instead, we first exploit the Rayleigh quotient to write it in terms of an optimization problem:

$$\begin{aligned}
 f(x) &= \lambda_{\min}(A(x)) = \min_{\|z\|_2=1} z^T A(x) z \\
 &= \min_{\|z\|_2=1} \sum_{i=1}^n z^T (x_i A_i) z \\
 &= \min_{\|z\|_2=1} \sum_{i=1}^n x_i (z^T A_i z) \\
 &= \min_{\|z\|_2=1} x^T g(z) \qquad g(z) \doteq [z^T A_1 z, \dots, z^T A_n z]^T \\
 \therefore -f(x) &= \max_{\|z\|_2=1} -x^T g(z).
 \end{aligned}$$

Observe that we have now defined the function $-f(x)$ at each point x as a maximum over a family of functions that are affine in x . This means that $-f$ is a convex function, thus f is **concave**.

- **Negative log of a concave function:** Recall that $\log x$ is **concave** on \mathbb{R}_+ . Therefore, $f(x) = -\log x$ is **convex** on \mathbb{R}_+ . Also, observe that f is nonincreasing. Suppose $g(y)$

is a concave function. By the composition rules, the function $f \circ g(y) = -\log(g(y))$ is convex on $\{y | g(y) > 0\}$.

- **Composing many things at once:** Consider the multiply-composed function

$$f(x) = -\log(-\log(\sum_{i=1}^m e^{a_i^\top x + b_i}))$$

on $\text{dom } f = \{x | \sum_{i=1}^m e^{a_i^\top x + b_i} < 1\}$. f is **convex**.

First, take as a given that $g(y) \doteq \log(\sum_{i=1}^m e^{y_i})$ is convex in y . (This is too complex to show directly with the tools we have—but may become an exercise in the future.)

The composition of g with an affine function, $g(Ax + b)$, is convex in x (because composition of a convex function with an affine function always preserves convexity). Letting the rows of A be a_i^\top , we have $g(Ax + b) = \log \sum_{i=1}^m e^{a_i^\top x + b_i}$. Then $-g(Ax + b)$ is concave in x .

As discussed in the previous bullet, the negative log of a concave function—which we have here—is a convex function on the domain where that concave function is positive.

The domain is then:

$$\begin{aligned} \text{dom } f &= \left\{ x \mid -\log \sum_{i=1}^m e^{a_i^\top x + b_i} > 0 \right\} \\ &= \left\{ x \mid \sum_{i=1}^m e^{a_i^\top x + b_i} < 1 \right\} \end{aligned}$$

We are not finished—we need the domain of f to be a convex set. This is not hard to show, however. First note that $\sum_{i=1}^m e^{a_i^\top x + b_i}$ is a convex function of x by the composition rules (exponential is convex, composition with affine preserves convexity). Hence the domain of f can be characterized as a sub-level set of $\sum_{i=1}^m e^{a_i^\top x + b_i}$, which for convex functions we know will be convex sets (see discussion 5).

11 Convex Optimization: Definitions

Required reading: Boyd. Ch 4, p 127 - 156.

A convex optimization problem is one where **the objective is a convex function, and the feasible set is a convex set.**

The standard form for convex optimization problems is

$$\begin{aligned} p^* &= \min_x f_0(x) \\ \text{s.t. } & f_i(x) \leq 0, \quad i = 1, \dots, p \\ & Ax = b, \end{aligned} \tag{2}$$

where f_0, \dots, f_p are convex functions.

Recall that the **sublevel set of a convex function is convex**, so the set

$$\mathcal{X}_i \doteq \{x \in \mathbb{R}^n : f_i(x) \leq 0\}$$

is convex when f_i is convex. Also, the set $\{x : Ax = b\}$ is an affine subspace of \mathbb{R}^n , which is convex. The intersection of convex sets is convex, so the total feasible set for (2), \mathcal{X} , which is defined by

$$\mathcal{X} = \mathcal{X}_1 \cap \mathcal{X}_2 \cap \dots \cap \mathcal{X}_p \cap \{x : Ax = b\},$$

is convex.

Remark. Notice that the equality constraint is required to be affine, rather than just a convex function. This is because, for convex function f , the set $\{x : f(x) = 0\}$ is **not necessarily convex**. For example, consider the function, $f(x) = \|x\|_2 - r$. The set $\{x : \|x\|_2 - r = 0\}$ defines the surface of an n -ball, but does not contain any of its interior points. Thus, this set is not convex.

11.1 Linear Programs (LP)

A **linear program (LP)** is an optimization problem with **linear objective and constraint functions**. Any linear program can be expressed in the form

$$\begin{aligned} p^* &= \min_x c^T x \\ \text{s.t. } & Gx \preceq h, \\ & Ax = b, \end{aligned} \tag{3}$$

for appropriate choices of c, A, b, G, h .

It is useful to interpret this geometrically: recall that the set

$$\mathcal{H}(g, h) \doteq \{x : g^T x \leq h\},$$

for vector g and scalar h , defines a **halfspace** (cuts R^n in half along a plane, and includes the plane and all the points to one side of it).

Thus, the set $\{x : Gx \preceq h\}$, for matrix G and vector h , specifies an intersection of halfspaces ($g_i^T x \leq h_i$, for $i = 1, \dots, m$, where g_i^T is the i th row of G). An intersection of halfspaces is a **polyhedron**.

The LP problem thus maximizes a linear objective over a polyhedron feasible set.

Fact 3. *Every linear program is a convex optimization problem.*

11.2 Quadratic Programs (QP)

A **quadratic program (QP)** is an optimization problem with a **convex quadratic objective function and affine constraint functions**. Any quadratic program can be expressed in the form

$$\begin{aligned} p^* = \min_x \quad & \frac{1}{2} x^T P x + q^T x \\ \text{s.t.} \quad & Gx \preceq h, \\ & Ax = b, \end{aligned} \tag{4}$$

for appropriate choices of $P \succeq 0, q, A, b, G, h$. Note that we require the objective function to be convex—hence we must have $P \succeq 0$. Because the objective is convex and the feasible set is convex (a polyhedron, as in LP), the QP is also convex.

If we have an optimization problem of this form, and $P \not\succeq 0$, we can call it a *nonconvex quadratic program*.

Remark. The factor of $1/2$ in the objective is arbitrary; we choose it to remove a factor of 2 in the gradient. (The gradient here is $\nabla_x f = Px + q$. If we did not include the factor of $1/2$, it would have been $\nabla_x f = 2Px + q$.)

12 Convex Optimization: Equivalent Optimization Problems

Required reading: Boyd. Ch 4, p 127 - 156.

12.1 Simple Sub-problems

You should recognize the following optimization problems, and their solutions, immediately (or at least quickly).

- Unconstrained, with linear objective:

$$p^* = \min_x f(x) \doteq a^T x = \begin{cases} -\infty & a \neq 0 \\ 0 & \text{otherwise} \end{cases}$$

- Unconstrained, with **convex** quadratic objective, i.e., $A \in \mathbb{S}_+^n$:

$$\begin{aligned} p^* &= \min_x f(x) \doteq \frac{1}{2} x^T A x + b^T x \\ &\implies \nabla_x f(x^*) = A x^* + b = 0 \\ &\implies x^* = -A^\dagger b && \text{using pseudo-inverse, } A^\dagger = V \Sigma^{-1} U^T \\ \therefore p^* &= -\frac{1}{2} b^T A^\dagger b && \text{plug in solution} \end{aligned}$$

- Scalar variable ($x \in \mathbb{R}$), linear objective, linear constraint ($x \geq c$, for some $c \in \mathbb{R}$):

$$p^* = \min_{x \geq c} a x = \begin{cases} -\infty & a < 0 \\ a c & \text{otherwise} \end{cases}$$

- Scalar variable, quadratic objective, linear constraint:

$$p^* = \min_{x \geq 0} \frac{1}{2} a x^2 - b x = \begin{cases} 0 & b/a \geq 0 \\ -\frac{b^2}{2a} & \text{otherwise} \end{cases}$$

To see this, consider the unconstrained solution ($x_u^* = b/a$). If $b/a \geq 0$, then the unconstrained solution—which is a global optimizer of the objective over all reals—optimizes the objective of the subset of reals, $x \geq 0$, and we have $x^* = x_u^*$. If $b/a < 0$, then the unconstrained optimum is to the left of 0. The function is *always non-decreasing* to the right of the minimum. Thus, the leftmost point in $\{x \geq 0\}$ must be optimal— $x^* = 0$ in this case. (See the “proof by picture” in Figure 1 if you don’t believe me.)

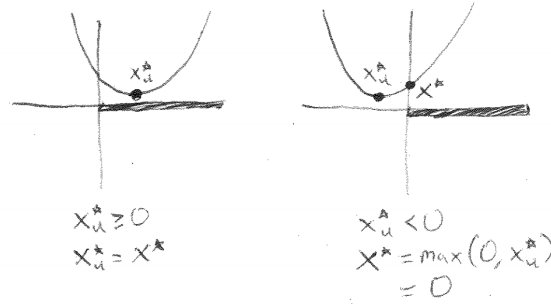


Figure 1: Optimizing a quadratic function with the constraint $x \geq 0$. The shaded part of the real line represents the feasible set. x_u^* is the global solution to the unconstrained problem. x^* solves the constrained problem.

- Vector variable ($x \in \mathbb{R}^n$), linear objective, linear constraint:

$$p^* = \left\{ \begin{array}{l} \min_x a^T x \\ \text{s.t. } b^T x \geq c \end{array} \right\} = \left\{ \begin{array}{ll} c\|a\|_2/\|b\|_2 & a = \gamma b, \text{ for } \gamma > 0 \\ -\infty & \text{otherwise} \end{array} \right.$$

Proof: Express x as a sum of two terms, one proportional to b , and one perpendicular to b (which we are allowed to do by fundamental theorem of linear algebra—think about it!). Notate this as $x = \alpha b + \nu$, with $\nu^T b = 0$. The problem then becomes:

$$\begin{aligned} p^* &= \min_x a^T x : b^T x \geq c \\ &= \min_{\alpha, \nu} a^T (\alpha b + \nu) : b^T (\alpha b + \nu) \geq c, \quad \nu^T b = 0 \\ &= \min_{\alpha, \nu} \alpha(a^T b) + a^T \nu : \alpha b^T b \geq c, \quad \nu^T b = 0 \\ &= \min_{\alpha \geq c/b^T b} \alpha(a^T b) + \min_{\nu: \nu^T b = 0} a^T \nu \end{aligned}$$

The first problem, minimizing over α , gives optimal value $-\infty$ if $a^T b < 0$; if $a^T b > 0$, its optimal point is $\alpha = c/b^T b$ and its optimal value is $c(a^T b)/(b^T b)$.

The second problem, minimizing over ν , gives optimal value $-\infty$ if $a \not\propto b$. If $a \propto b$, the optimal value is 0 by the constraint.

Thus, to have a finite value, we must have $a \propto b$ and $a^T b > 0$; this is achieved when they are co-linear ($a = \gamma b$ for $\gamma \geq 0$).

How should I be able to see this immediately? This is a problem where you're minimizing a linear function over a halfspace. If the direction of decrease has any component that points *into* the halfspace, then the objective clearly goes to $-\infty$. So the direction of decrease has to point directly out of the halfspace, and then the

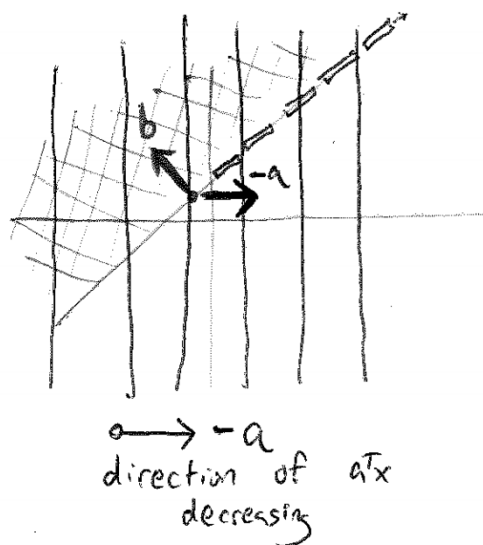


Figure 2: Optimizing a linear function, $a^T x$, over a halfspace $b^T x \geq c$. The objective decreases in the direction $-a$. If any vector in the hyperplane $b^T x = c$ has positive inner product with $-a$, then the objective decreases as you move along the hyperplane (following the dashed path), and the optimal value is $-\infty$. In order for the optimization problem to have finite value, a must align with b .

function is minimized at any point on the hyperplane. (Getting the actual value of the function is just algebra at this point.) If this is still unclear, try Figure 2.

Even though your geometric intuition should point you in the right direction², you should still explain the optimization approach to the solution. (Rigor is good!)

- Rayleigh quotient problem: for $A \in \mathbb{S}^n$,

$$\lambda_{\max}(A) = \max_x x^T A x : \|x\|_2 = 1.$$

The proof involves duality, which you haven't learned yet, but you should know this one stone-cold.

Fact 4. *The Rayleigh quotient problem is nonconvex, even when $A \succeq 0$.*

The constraint region is the surface of a hypersphere—but does not contain the interior of the hypersphere. (Think about it: is the boundary of a circle a convex set? It has a hole, so it is not.)

²yes this was a pun. sorry not sorry

- Relaxed Rayleigh quotient problem: for $A \in \mathbb{S}^n$,

$$\max(0, \lambda_{\max}(A)) = \max_x x^T A x \quad : \quad \|x\|_2 \leq 1.$$

In this problem, we have *relaxed* the constraint, which allows more points in the feasible set. There are two cases here: $\lambda_{\max}(A) > 0$, and $\lambda_{\max}(A) \leq 0$. In the first case, the optimum value is $\lambda_{\max}(A)$ (to see this, compare with the original Rayleigh quotient problem). In the latter case, A is negative-definite, so for any $x \neq 0$, $x^T A x < 0$. Thus, the objective is optimized with $x^* = 0 \implies x^{*T} A x^* = 0$. The $\max(0, \lambda_{\max}(A))$ summarizes.

We could also take a different approach that involves manipulating (reparametrizing) the variables of the optimization problem. Note that

$$\{x \mid \|x\|_2 \leq 1\} = \{\alpha z \mid \alpha \in [0, 1], z \in \mathbb{R}^n, \|z\|_2 = 1\}.$$

This allows us to rewrite the problem as

$$\begin{aligned} \max_{\|x\|_2 \leq 1} x^T A x &= \max_{\substack{0 \leq \alpha \leq 1 \\ \|z\|_2 = 1}} (\alpha z)^T A (\alpha z) \\ &= \max_{\substack{0 \leq \alpha \leq 1 \\ \|z\|_2 = 1}} \alpha^2 z^T A z \\ &= \max_{0 \leq \alpha \leq 1} \alpha^2 \left(\max_{\|z\|_2 = 1} z^T A z \right) \\ &= \max_{0 \leq \alpha \leq 1} \alpha^2 \lambda_{\max}(A) \\ &= \begin{cases} 0 & \lambda_{\max}(A) < 0 \\ \lambda_{\max}(A) & \text{otherwise.} \end{cases} \end{aligned}$$

12.2 Equality-Preserving Transformations

- **Epigraph form:** The following is trivially true, for any $R \in \mathbb{R}$:

$$\begin{aligned} R &= \min_t t \\ \text{s.t. } t &\geq R. \end{aligned}$$

Consequently, we are justified in putting an optimization program in *epigraph form*,

$$\begin{aligned} \min_x f(x) &= \min_{x,t} t \\ \text{s.t. } t &\geq f(x). \end{aligned}$$

Note that this transformation preserves convexity: if f is a convex function of f , the constraint $t \geq f(x)$ is jointly convex in x and t . Furthermore, the objective is convex (linear) in the variables of optimization. So, the epigraph form of a convex optimization problem is itself a convex optimization problem.

- **Domain-splitting:** For any problem of the form

$$p^* = \min_{x \in \mathcal{X}} f(x),$$

if we can express $\mathcal{X} = \mathcal{Y} \cup \mathcal{Z}$ for some sets \mathcal{Y}, \mathcal{Z} , then

$$p^* = \min \left(\min_{x \in \mathcal{Y}} f(x), \min_{x \in \mathcal{Z}} f(x) \right).$$

- **Reparametrization:** By fundamental theorem of linear algebra, for any vector $x \in \mathbb{R}^n$ and matrix $A \in \mathbb{R}^{n \times m}$, we can write $x = Az + v$ for some $z \in \mathbb{R}^m$ and $v \in \mathcal{N}(A^T)$. So the following optimization problems are equivalent:

$$\begin{aligned} p^* &= \min_{x \in \mathcal{X}} f(x) \\ &= \min_{z, v} f(Az + v) : Az + v \in \mathcal{X}, A^T v = 0. \end{aligned}$$

- **Composing the objective with monotone-increasing functions:** Suppose $\phi : \mathbb{R} \rightarrow \mathbb{R}$ is monotone increasing. Then for any x, y , $f(x) \geq f(y) \iff \phi(f(x)) \geq \phi(f(y))$. Thus, the following two problems have the same *optimal sets*:

$$\arg \min_{x \in \mathcal{X}} f(x) = \arg \min_{x \in \mathcal{X}} \phi(f(x)),$$

and the optimal values are related by

$$\phi \left(\min_{x \in \mathcal{X}} f(x) \right) = \min_{x \in \mathcal{X}} \phi(f(x)).$$

12.3 Constraint Magic

In this section, we deal with common transformations of constraints that you should know about.

- **Max-forall trick:** Suppose you have a constraint of the form

$$0 \geq \max_{x \in \mathcal{X}} f(x). \tag{5}$$

Recall that $\forall y \in \mathcal{X}, \max_{x \in \mathcal{X}} f(x) \geq f(y)$. Thus, (5) *implies* (we're only showing one direction with this step) that

$$\forall x \in \mathcal{X}, \quad 0 \geq f(x). \quad (6)$$

Suppose x^* maximizes f in \mathcal{X} . Then (6) implies that $0 \geq f(x^*) = \max_{x \in \mathcal{X}} f(x)$, so the implication goes both ways. We conclude that **any time you see a constraint on a maximum of a function over a set, you can replace it with multiple constraints, one for each element in the set:**

$$0 \geq \max_{x \in \mathcal{X}} f(x) \iff \forall x \in \mathcal{X}, \quad 0 \geq f(x).$$

- **Absolute value trick:** Suppose you have a constraint of the form

$$|x_i| \leq s_i,$$

for variable x_i . This can be rewritten as two inequalities,

$$|x_i| \leq s_i \iff \begin{aligned} x_i &\leq s_i, \\ -x_i &\leq s_i. \end{aligned}$$

The feasible set for x_i is **convex** in this case: $-s_i \leq x_i \leq s_i$.

CAVEAT: this gets you a convex feasible set (works the way you want it to) when the absolute value of something is bounded ABOVE. If the absolute value is bounded below, you may get a non-convex feasible set. If the constraint is an equality constraint, you definitely get a non-convex feasible set.

Observe: $|x_i| \geq s_i$ implies $x_i \geq s_i$ or $x_i \leq -s_i$. *This set is not convex for x_i for arbitrary s_i .* Similarly, $|x_i| = s_i$ implies $x_i \in \{s_i, -s_i\}$; this set is also not convex for x_i . See Figure 3.

Example 7. Consider an optimization problem of the form

$$\min_x a^T x + \|x\|_\infty.$$

We will show that through constraint magic, we can express this problem as an LP. We proceed as follows:

$$\begin{aligned} p^* &= \min_x a^T x + \|x\|_\infty \\ &= \min_{x,s} a^T x + s \quad : \quad s \geq \|x\|_\infty && \text{epigraph form of } \|x\|_\infty \\ &= \min_{x,s} a^T x + s \quad : \quad s \geq \max_i |x_i| && \text{definition of } \|x\|_\infty \\ &= \min_{x,s} a^T x + s \quad : \quad \forall i, \quad s \geq |x_i| && \text{max-forall trick} \\ &= \min_{x,s} a^T x + s \quad : \quad \forall i, \quad -s \leq x_i \leq s. && \text{absolute value trick} \end{aligned}$$

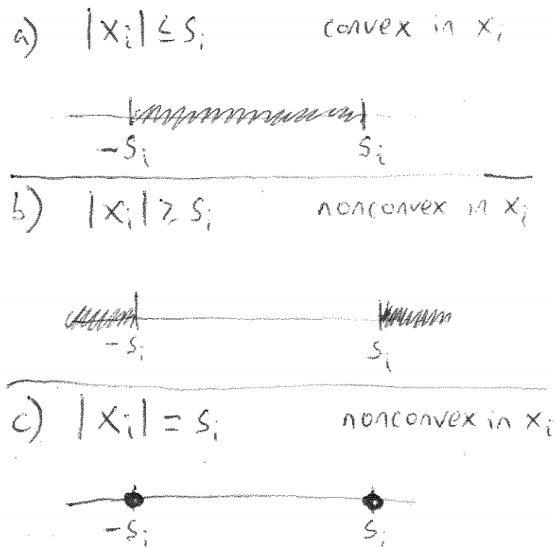


Figure 3: Geometric intuition for constraints involving absolute values. Here, we are considering a case where $s_i \geq 0$.

In the final step, the objective is linear in (x, s) and the constraints are also linear in x, s . (If you have trouble seeing how it's linear in both x and s , observe that you can write $\forall i, s - x_i \geq 0, s + x_i \geq 0$.)

13 Duality Intro: Defining Weak and Strong Duality

In this section, we begin to explore **duality**, a subject where we study the relationships between the optimization problems

$$\min_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} f(x, y),$$

which is called a min-max problem, and

$$\max_{y \in \mathcal{Y}} \min_{x \in \mathcal{X}} f(x, y),$$

which is called a max-min problem.

Although the difference may seem subtle, it is crucial: many problems are easier to solve in one formulation than the other, so being able to transform freely between them is highly beneficial.

Furthermore, the special case of **Lagrangian duality** allows us to make a wide range of constrained optimization problems analytically solvable.

13.1 Weak Duality

Proposition. The principle of **weak duality** is as follows: for every set \mathcal{X} , set \mathcal{Y} , and function $f : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$, the following inequality holds:

$$p^* \doteq \min_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} f(x, y) \geq \max_{y \in \mathcal{Y}} \min_{x \in \mathcal{X}} f(x, y) \doteq d^*. \quad (7)$$

The difference $p^* - d^*$ is called the **duality gap**.

Proof. First, the definitions of optimality give us the following two inequalities:

$$\begin{aligned} \forall x \in \mathcal{X}, y \in \mathcal{Y}, \quad & \max_{\bar{y} \in \mathcal{Y}} f(x, \bar{y}) \geq f(x, y), \\ \forall x \in \mathcal{X}, y \in \mathcal{Y}, \quad & f(x, y) \geq \min_{\bar{x} \in \mathcal{X}} f(\bar{x}, y). \end{aligned}$$

It follows immediately that

$$\forall x \in \mathcal{X}, \forall y \in \mathcal{Y}, \quad \max_{\bar{y} \in \mathcal{Y}} f(x, \bar{y}) \geq \min_{\bar{x} \in \mathcal{X}} f(\bar{x}, y).$$

We use the forall—max equivalence to get:

$$\forall x \in \mathcal{X}, \quad \max_{\bar{y} \in \mathcal{Y}} f(x, \bar{y}) \geq \max_{y \in \mathcal{Y}} \min_{\bar{x} \in \mathcal{X}} f(\bar{x}, y).$$

Because this holds for all x , it holds for the x that minimizes the left-hand side; thus the proof is complete. \square

Weak duality *always holds*, and can come in handy sometimes, even though we'll generally prefer strong duality.

13.2 Strong Duality

We say that strong duality holds for a min-max problem when

$$\min_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} f(x, y) = \max_{y \in \mathcal{Y}} \min_{x \in \mathcal{X}} f(x, y).$$

Strong duality does not generally hold. If you want to claim that it does, you have to justify it in some sense. We will talk about the conditions for strong duality to hold later in the notes.

14 Duality Intro: Lagrange Dual Problems

In this section, we deal with **Lagrange duality**. In Lagrange duality, we study constrained **primal problems** of the form

$$\begin{aligned} p^* = \min_x \quad & f_0(x) \\ \text{s.t.} \quad & f_i(x) \leq 0, \quad i = 1, \dots, m \\ & h_i(x) = 0, \quad i = 1, \dots, p. \end{aligned} \tag{8}$$

This constrained primal problem is equivalent to the following min-max optimization problem:

$$p^* = \min_x \max_{\lambda \succeq 0, \nu} f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) + \sum_{j=1}^p \nu_j h_j(x). \tag{9}$$

Here, we interpret the **dual variables** λ, ν to be the Lagrange multipliers (which you might remember from calculus). Observe that the optimization over x is now *unconstrained*: the constraints have been added to the objective as penalty terms, where a violation receives an infinite penalty.

The optimization objective in (9) has a special name: we call it the Lagrangian, \mathcal{L} , and it is a function of both the primal and dual variables:

$$\mathcal{L}(x, \lambda, \nu) \doteq f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) + \sum_{j=1}^p \nu_j h_j(x).$$

The act of moving a constraint function into the objective by Lagrange multipliers is called *dualizing* the constraint.

A lower bound on p^* can be found by switching the min and the max:

$$p^* \geq d^* \doteq \max_{\lambda \succeq 0, \nu} \min_x f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) + \sum_{j=1}^p \nu_j h_j(x). \tag{10}$$

That this is a lower bound follows from the principle of weak duality.

The function which is obtained by solving the inner optimization problem over x , for fixed λ, ν , is called the **Lagrange dual function**, and it is usually denoted by $g(\lambda, \nu)$:

$$g(\lambda, \nu) \doteq \min_x f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) + \sum_{j=1}^p \nu_j h_j(x).$$

The problem of optimizing the Lagrange dual function over the dual variables is called a **Lagrange dual problem**, or simply the dual problem:

$$d^* = \max_{\lambda \succeq 0, \nu} g(\lambda, \nu).$$

14.1 Why Does the Min-Max Formulation Work?

In this section, we aim to verify that the primal problem shown in (8) is equivalent to the min-max problem shown in (9).

We will begin from the min-max form (9), and solve the inner maximization problems over λ, ν .

We proceed by first exploiting the fact that the optimization over λ, ν is *decoupled*:

$$\begin{aligned} & \min_x \max_{\lambda \succeq 0, \nu} f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) + \sum_{j=1}^p \nu_j h_j(x) \\ &= \min_x \left(f_0(x) + \sum_{i=1}^m \left(\max_{\lambda_i \geq 0} \lambda_i f_i(x) \right) + \sum_{j=1}^p \left(\max_{\nu} \nu_j h_j(x) \right) \right). \end{aligned}$$

Now, let's examine the sub-problems over the λ 's and the ν 's. **Notice that the optimization over λ is constrained to $\lambda \succeq 0$, but no such constraint exists over ν . This is important.**

The sub-problem over λ_i :

$$\max_{\lambda_i \geq 0} \lambda_i f_i(x) = \begin{cases} 0 & f_i(x) \leq 0 \\ \infty & f_i(x) > 0. \end{cases}$$

The sub-problem over ν_j :

$$\max_{\nu_j \geq 0} \nu_j h_j(x) = \begin{cases} 0 & h_j(x) = 0 \\ \infty & h_j(x) \neq 0. \end{cases}$$

Adding up the contributions from all the different terms, we get

$$\begin{aligned} & \min_x \left(f_0(x) + \sum_{i=1}^m \left(\max_{\lambda_i \geq 0} \lambda_i f_i(x) \right) + \sum_{j=1}^p \left(\max_{\nu} \nu_j h_j(x) \right) \right) \\ &= \min_x \begin{cases} f_0(x) & \text{if } f_i(x) \leq 0, \text{ for } i = 1, \dots, m, \quad h_j(x) = 0, \text{ for } j = 1, \dots, p, \\ \infty & \text{otherwise.} \end{cases} \end{aligned}$$

Suppose that there is any point x which satisfies $f_i(x) \leq 0$ for all i and $h_j(x) = 0$ for all j , and for which $f_0(x)$ is well-defined (takes a finite value). Then that point is strictly better than any point which does not satisfy all of those constraints (because in those cases, the penalized objective has value infinity). Thus we can remove those points from the feasible set without changing the optimal point or value: that is,

$$\begin{aligned} \min_x \begin{cases} f_0(x) & \text{if } f_i(x) \leq 0, \text{ for } i = 1, \dots, m, \quad h_j(x) = 0, \text{ for } j = 1, \dots, p, \\ \infty & \text{otherwise.} \end{cases} \\ = \min_x f(x) \quad : \quad f_i(x) \leq 0 \text{ for } i = 1, \dots, m, \quad h_j(x) = 0 \text{ for } j = 1, \dots, p. \end{aligned}$$

Intuitively, **if you replace constraints with penalty functions in the objective, and you make the penalty infinitely steep for violating the constraint, your penalized formulation is equivalent to your constrained formulation.**

14.2 Lagrange Dual Problems Always Exist

Regardless of whether or not strong duality holds (that is, regardless of whether or not $p^* = d^*$), **we can always form a Lagrange dual problem.**

14.3 Lagrange Dual Problems Are Not Unique

There can be many equivalent ways to express the same set of constraints, resulting in many different Lagrange dual problems for the same primal problem. Some of these formulations are more useful than others.

For example, let's study a very simple case where things quickly go *horribly wrong* if we make a bad choice of how to represent and dualize the constraints.

Example 8. The primal problem under consideration is

$$p^* = \min_x \frac{1}{2} \|x\|_2^2 + b^T x \quad : \quad (c^T x)^2 \leq 1. \quad (11)$$

Dualizing the Constraint As-Is:

The constraint is simple-looking, so dualizing it shouldn't be too hard. What happens when we try to dualize it, as-is, and find a dual problem?

We proceed:

$$\begin{aligned}
p^* &= \min_x \frac{1}{2} \|x\|_2^2 + b^T x \quad : \quad (c^T x)^2 \leq 1 \\
&= \min_x \max_{\lambda \geq 0} \frac{1}{2} \|x\|_2^2 + b^T x + \frac{1}{2} \lambda ((c^T x)^2 - 1) \\
&\geq \max_{\lambda \geq 0} \min_x \frac{1}{2} \|x\|_2^2 + b^T x + \frac{1}{2} \lambda ((c^T x)^2 - 1) \\
&\doteq d^*.
\end{aligned}$$

At this stage, we rewrite the objective, with $\|x\|_2^2 = x^T x$ and $(c^T x)^2 = x^T (cc^T) x$, and try to solve for the dual function

$$g(\lambda) = \min_x \mathcal{L}(x, \lambda), \quad \text{where } \mathcal{L}(x, \lambda) \doteq \frac{1}{2} x^T (I + \lambda cc^T) x + b^T x - \lambda.$$

Note that this is *unconstrained, convex, quadratic* optimization, which is easy—set the gradient to zero and solve. We find:

$$\nabla_x \mathcal{L}(x, \lambda) = (I + \lambda cc^T)x + b = 0 \implies x^*(\lambda) = -(I + \lambda cc^T)^{-1}b.$$

After plugging this $x^*(\lambda)$ back in, we obtain the dual function

$$g(\lambda) = -\frac{1}{2} b^T (I + \lambda cc^T)^{-1} b - \lambda,$$

and the corresponding dual problem,

$$d^* = \max_{\lambda \geq 0} -\frac{1}{2} b^T (I + \lambda cc^T)^{-1} b - \lambda.$$

This is a horrible, awful dual problem that is functionally useless to us. Observe how λ appears in it: it shows up inside a matrix inverse. This is a super complicated dependence that makes things almost totally analytically intractable. Bad. Avoid this.

Dualizing Equivalent Constraints Instead:

Observe that $(c^T x)^2 \leq 1$ is equivalent to $-1 \leq c^T x \leq 1$. Instead of dualizing the original quadratic constraint, let's dualize the equivalent linear constraints and see what happens.

The problem becomes:

$$\begin{aligned}
p^* &= \min_x \frac{1}{2} \|x\|_2^2 + b^T x \quad : \quad -1 \leq c^T x \leq 1 \\
&= \min_x \max_{\lambda_1, \lambda_2 \geq 0} \frac{1}{2} \|x\|_2^2 + b^T x + \lambda_1 (c^T x - 1) + \lambda_2 (-1 - c^T x) \\
&\geq \max_{\lambda_1, \lambda_2 \geq 0} \min_x \frac{1}{2} \|x\|_2^2 + b^T x + \lambda_1 (c^T x - 1) + \lambda_2 (-1 - c^T x) \\
&\doteq d^*.
\end{aligned}$$

We obtain the dual function the same way as before—by solving the inner optimization over x , by setting the gradient (wrt x) of the Lagrangian equal to zero and solving:

$$\nabla_x \mathcal{L}(x, \lambda_1, \lambda_2) = x^* + b + \lambda_1 c - \lambda_2 c = 0 \implies x^*(\lambda_1, \lambda_2) = -b - \lambda_1 c + \lambda_2 c.$$

After plugging this $x^*(\lambda_1, \lambda_2)$ back in, we obtain the dual function:

$$g(\lambda) \doteq -\frac{1}{2} (b + \lambda_1 c - \lambda_2 c)^T (b + \lambda_1 c - \lambda_2 c) - \lambda_1 - \lambda_2,$$

which we can see is *quadratic* in λ_1, λ_2 - this is much nicer than before.

14.4 Lagrange Dual Problems for Maximization Primal Problems

Sometimes, the primal problem is a maximization problem and not a minimization problem. This poses no issue to forming a Lagrange dual—which will in this case turn out to be a minimization over the dual variables rather than a maximization—but we do have to be careful about signs.

The way we dualize constraints in this case is as follows:

$$\begin{aligned}
\max_x f_0(x) \quad : \quad & f_i(x) \leq 0 \text{ for } i = 1, \dots, m, \quad h_j(x) = 0 \text{ for } j = 1, \dots, p. \\
= \max_x \min_{\lambda \geq 0, \nu} & f_0(x) - \sum_{i=1}^m \lambda_i f_i(x) + \sum_{j=1}^p \nu_j h_j(x).
\end{aligned}$$

Note that we have a minus sign in front of λ_i now, unlike before. This is important.

Minor exercise. Verify that this max-min problem is equivalent to the original max problem by showing that the inner minimization problem has the solution

$$\begin{cases} f_0(x) & \text{if } f_i(x) \leq 0 \text{ for } i = 1, \dots, m, \quad h_j(x) = 0 \text{ for } j = 1, \dots, p, \\ -\infty & \text{otherwise.} \end{cases}$$

The dual function in this case is

$$g(\lambda, \nu) = \max_x f_0(x) - \sum_{i=1}^m \lambda_i f_i(x) + \sum_{j=1}^p \nu_j h_j(x),$$

and the dual problem is

$$\min_{\lambda \geq 0, \nu} g(\lambda, \nu).$$

15 Duality Intro: When Does Strong Duality Hold?

In this section, we will give two theorems for when strong duality holds. These are Slater's theorem, which gives a simple condition for strong duality of Lagrange dual problems, and Sion's minimax theorem, which gives very general conditions under which strong duality is attained.

Because these theorems concern min-max problems, they are called **minimax theorems**.

15.1 Strong Duality for Lagrange Dual Problems

Theorem 3 (Slater's Theorem). *Suppose the primal problem,*

$$\begin{aligned} p^* &= \min_x f_0(x) \\ \text{s.t. } & f_i(x) \leq 0, \quad i = 1, \dots, m \\ & h_i(x) = 0, \quad i = 1, \dots, p. \end{aligned}$$

is a convex optimization problem. Let \mathcal{D} denote the domain of the problem, $\mathcal{D} \doteq \text{dom}(f) \cap \bigcap_{i=1}^m \text{dom}(f_i) \cap \bigcap_{i=1}^n \text{dom}(h_i)$. Suppose that there exists an $x \in \text{relint } \mathcal{D}$ such that

$$f_i(x) < 0, \quad i = 1, \dots, m, \quad h_i(x) = 0, \quad i = 1, \dots, n.$$

Then strong duality holds, in that, for the Lagrangian

$$\mathcal{L}(x, \lambda, \nu) = f(x) + \sum_{i=1}^m \lambda_i f_i(x) + \sum_{i=1}^n \nu_i h_i(x),$$

we have

$$p^* = \min_x \max_{\lambda \geq 0, \nu} \mathcal{L}(x, \lambda, \nu) = \max_{\lambda \geq 0, \nu} \min_x \mathcal{L}(x, \lambda, \nu) = d^*.$$

A weaker form of this holds if the first k inequality constraints are affine: then there only needs to exist an $x \in \text{relint } \mathcal{D}$ such that

$$f_i(x) \leq 0, \quad i = 1, \dots, k, \quad f_i(x) < 0, \quad i = k+1, \dots, m, \quad h_i(x) = 0, \quad i = 1, \dots, n.$$

for strong duality to hold.

Note that **Slater's theorem only works for Lagrangians that you build by dualizing constraints**. This cannot be applied to arbitrary min-max problems: you need to have the primal constrained problem to make this work.

15.2 Strong Duality for General Min-Max Problems

Theorem 4 (Sion's Minimax Theorem). *For a function $\mathcal{L} : X \times Y \rightarrow \mathbb{R}$, if X is a compact and convex subset of a linear topological space and Y is a convex subset of a linear topological space, and*

- $\forall x \in X, \mathcal{L}(x, \cdot)$ is quasiconcave and upper semi-continuous, and
- $\forall y \in Y, \mathcal{L}(\cdot, y)$ is quasiconvex and lower semi-continuous,

then

$$\min_{x \in X} \sup_{y \in Y} \mathcal{L}(x, y) = \sup_{y \in Y} \min_{x \in X} \mathcal{L}(x, y).$$

Notes on Sion's theorem:

- Don't worry about what a linear topological space is. \mathbb{R}^n (with a normal dot product) is a linear topological space and that is the only one you need to care about.
- Quasiconcavity and quasiconvexity are sort-of generalizations of concavity and convexity. All concave functions are quasiconcave (but not vice-versa), and all convex functions are quasiconvex (but not vice-versa).
- If something is continuous it is also both upper and lower semi-continuous.
- A compact set contains its limit points. The open interval $(a, b) \subset \mathbb{R}$ is not compact. The closed interval $[a, b] \subset \mathbb{R}$ is compact. The set $\{x : \|x\|_p \leq 1\}$ is compact.
- You have to satisfy conditions on both the Lagrangian function \mathcal{L} , and the feasible sets X, Y in order to apply Sion's theorem.

If you see a Lagrangian function $\mathcal{L}(x, y)$ that is concave in y , convex in x , and continuous, it satisfies the conditions of Sion's theorem for the Lagrangian.

If both feasible sets X and Y are convex, and at least one of them is compact, that's sufficient for Sion's theorem.

We have experienced some difficulty with the compactness condition in some problems—if this is an issue in your homework and you can't prove strong duality any other way, don't worry about this hiccup. There is probably some other theorem that can save you.

16 KKT Conditions

The Karush-Kuhn-Tucker (KKT) conditions characterize optimality for primal problem / Lagrange dual problem pairs when all functions involved are differentiable with respect to x . **These only work for problems that have the Lagrange duality structure. Not for arbitrary min-max problems.**

Let x^* and (λ^*, ν^*) be any primal and dual optimal points with zero duality gap (that is, $p^* = d^*$). Any such points must satisfy

$$\begin{aligned} 0 &\geq f_i(x^*) \quad i = 1, \dots, m \\ 0 &= h_i(x^*) \quad i = 1, \dots, n \\ 0 &\preceq \lambda^* \\ 0 &= \lambda_i^* f_i(x^*) \quad i = 1, \dots, m \\ 0 &= \nabla_x f_0(x^*) + \sum_{i=1}^m \lambda_i^* \nabla f_i(x^*) + \sum_{i=1}^n \nu_i^* \nabla h_i(x^*). \end{aligned}$$

When the primal problem is convex, the KKT conditions are also sufficient conditions for the primal and dual points to be optimal: if x^* and (λ^*, ν^*) are any points that satisfy the KKT conditions, then they are primal and dual optimal for that problem.

Notes on the KKT conditions:

- The first three conditions just require x^*, λ^* to be primal and dual feasible, respectively.
- The last condition comes from solving the unconstrained minimization problem over x : to compute $\min_x \mathcal{L}(x, \lambda^*, \nu^*)$, we set the gradient to zero:

$$\nabla_x \mathcal{L}(x^*, \lambda^*, \nu^*) = \nabla_x f_0(x^*) + \sum_{i=1}^m \lambda_i^* \nabla f_i(x^*) + \sum_{i=1}^n \nu_i^* \nabla h_i(x^*) = 0.$$

- The fourth condition, $\lambda_i^* f_i(x^*) = 0$, is called **complementary slackness**. This is an interesting and subtle condition, but proving it is straightforward. Suppose $f_i(x^*) < 0$. What is

$$\lambda_i^* = \arg \max_{\lambda_i \geq 0} \lambda_i f_i(x^*)?$$

Spoiler alert: $\lambda_i^* = 0$. Thus, for this case, $\lambda_i^* f_i(x^*) = 0$. What about in the case where $f_i(x^*) = 0$? This case seems to solve itself. The case $f_i(x^*) > 0$ is not possible (it violates a feasibility condition for x^*), so we don't have to worry about it.

17 More Magic

In this section, I'll go through a few magic tricks that don't specifically fit anywhere else, but are neat and will add to your understanding / problem-solving ability:

- the norm trick,
- the $\min \Leftrightarrow \exists$ trick,
- and the business of when to turn a bad case of the infinities into a constraint.

17.1 The Norm Trick

Calling this a trick is maybe somewhat misleading. So first, let's get on solid ground. Given a norm $\|\cdot\|$, we define the dual norm $\|\cdot\|_*$ by:

$$\|x\|_* \doteq \max_z z^T x \quad : \quad \|z\| \leq 1.$$

The “trick” is that the p -norm and the q -norm, for $1/p + 1/q = 1$, are dual norms to each other: that is,

$$\|x\|_p = \max_z z^T x \quad : \quad \|z\|_q \leq 1,$$

where $1/p + 1/q = 1$, and $\|\cdot\|_p$ is the l_p -norm, defined by $\|z\|_p = (\sum_i z_i^p)^{1/p}$.

Example 9. Consider the problem

$$p^* = \min_{x \in X} f(x) + \lambda \|x\|_2. \tag{12}$$

The norm “trick” allows us to rewrite the l_2 -norm regularization in (12) as

$$p^* = \min_{x \in X} \max_z f(x) + \lambda z^T x \quad : \quad \|z\|_2 \leq 1.$$

If f is convex and X is a convex subset of \mathbb{R}^n , Sion's theorem tells us that strong duality holds. (Note that we have compactness on the z -set.)

17.2 “Min—There Exists” Trick

In the special review session, a robust-constraint problem came up with the form

$$\min_{x \in \mathcal{X}} f(x) : \left(\min_{z \in \mathcal{Z}} z^T x \right) \leq b.$$

The interesting thing is that this problem is equivalent to

$$\min_{x \in \mathcal{X}, z \in \mathcal{Z}} f(x) : z^T x \leq b.$$

In this latter form, we’ve lifted up the variable z from only existing in a sub-problem in the constraint, to being a variable in the overall optimization. How did we do this?

It’s conceptually similar to the max—forall trick. The equivalence is this:

$$\left(\min_{z \in \mathcal{Z}} z^T x \right) \leq b \iff \exists z \in \mathcal{Z} : z^T x \leq b.$$

Proving this is straightforward. First, consider the forward direction. If $(\min_{z \in \mathcal{Z}} z^T x) \leq b$, then there is some z (perhaps the minimizer) such that $z^T x \leq b$.

Consider the reverse direction. Suppose there is some $\bar{z} \in \mathcal{Z}$ such that $\bar{z}^T x \leq b$. We know that by definition, $\min_{z \in \mathcal{Z}} z^T x \leq \bar{z}^T x$, so it follows transitively that $\min_{z \in \mathcal{Z}} z^T x \leq b$. QED.

Note that **if a variable in an optimization problem does not appear in the objective, but does appear in a constraint, then solving for that variable is simply a feasibility problem.**

17.3 Bad Case of the Infinities

The following statements are always true as long as \mathcal{X} is not empty and $f(x)$ is well-defined everywhere on \mathcal{X} :

$$\begin{aligned} \min_x \begin{cases} f(x) & x \in \mathcal{X} \\ \infty & \text{otherwise} \end{cases} \\ = \min_x f(x) \quad : \quad x \in \mathcal{X}, \end{aligned}$$

and

$$\begin{aligned} \max_x \begin{cases} f(x) & x \in \mathcal{X} \\ -\infty & \text{otherwise} \end{cases} \\ = \max_x f(x) \quad : \quad x \in \mathcal{X}. \end{aligned}$$

That is, if you are optimizing a function with multiple cases, and one of the cases drives the function to the wrong end of infinity ($+\infty$ if you are minimizing, and $-\infty$ if you are maximizing), then you can turn the unconstrained optimization problem into an equivalent constrained problem by turning the condition into a constraint.