Lecture Notes on Convex Optimization

Shashank Vatedka
IIT Hyderabad

January 12, 2024

Contents

Preface

These are lecture notes for the course EE5606 Convex Optimization taught at IIT Hyderabad.

This is a working document, and will be updated constantly. If you find any errors, please notify the instructor.

Chapter 1

Introduction

Every problem where there is a notion of a "best" solution can be posed formally as an optimization problem. Essentially, if the problem has multiple solutions, and there is a way to quantify how good the solution is, then this can be formulated as a mathematical optimization problem. Almost every engineering problem can be cast as an optimization problem.

- In processor design, we want to pack the maximum number of transistors (get as much compute power as possible), while ensuring the power consumption is as low as possible.
- In wireless communication, we want to design a system such that we can transmit information at as high a rate as possible, using as minimum resources (power, bandwidth) as possible, while maintaining certain quality-of-service constraints.
- In signal denoising, we want to obtain as faithful a representation as possible to the original signal from a noisy version of the signal.
- In an object detection problem (image processing/computer vision), we want to detect objects (cats/cars/buildings/humans) as reliably as possible.
- In portfolio optimization (finance), we want to invest capital in a set of assets so as to get the highest returns.
- In an industrial control problem, we want to control the input so that the system behavior is as close to the desired performance as possible.

The applications are endless.

Given any engineering problem, the first step is to generally formulate a mathematical model that enables us to pose this as a mathematical optimization problem. This is based on domain knowledge and data, and is only an approximation of the (physical) problem.

1.0.1 The general form of an optimization problem

In any optimization problem, there are three objects:

- The *optimization variable(s)*, or the parameter(s) that we can vary

- An *objective function*, which measures how good a particular solution is
- A set of *constraints*, which model the physical/logistical limitations under which we can vary the optimization variable

For the examples given above, try to identify what might be the optimization variables, the objective function, and the set of constraints.

Definition 1.0.1: Constrained optimization problem

Given functions $f : \mathbb{R}^n \to \mathbb{R}$, and $g : \mathbb{R}^n \to \mathbb{R}^m$, find

$$\underline{x}^* = \arg \min_{g(\underline{x}) \geqslant \underline{0}} f(\underline{x}).$$

Here, $\underline{x}$ is the vector of optimization variables, f is called the objective function, and g is the constraint.

The first step in solving any engineering problem is to obtain a mathematical formulation of the problem. In many cases, this is more of an art, and once we have a well-formulated optimization problem, the job is half-done.

Problem 1.0.1: General form

Can we formulate

$$\underline{x}^* = \arg \max_{g(\underline{x}) \geqslant \underline{0}} f(\underline{x})$$

in the above form? What if some of the constraints are of the form $g_i(\underline{x}) \geqslant a_i$, and some of the form $g_i(x) \leqslant b_i$? What if we have constraints of the form $g_i(\underline{x}) > a_i$?

Unfortunately, many mathematical problems of interest may have optimal solutions, but finding this optimal solution may be computationally hard. We then try to find algorithms that approximately solve our optimization problem. However, as we will see later in this course, there are certain classes of problems for which we can find the optimal solution *efficiently*.

1.1 Examples of constrained optimization problems

1.1.1 Least squares solution for a system of linear equations

Suppose that we have a system of linear equations

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

where $\underline{x} \in \mathbb{R}^n$, $\underline{b} \in \mathbb{R}^k$, and A is a $k \times n$ full-rank matrix. If $k > n$, then clearly the system either has a unique solution, or no solution at all. It therefore makes sense to look for a vector $\underline{x}$ that best approximates the system, i.e., finding

$$\underline{x}^* = \arg \min_{\underline{x} \in \mathbb{R}^n} \|A\underline{x} - b\|_2^2.$$

This is what we call the least-squares solution. This is an example of an unconstrained optimization problem (as we will see later, it is also convex). Here, the objective function is $f(x) = \|A\underline{x} - \underline{b}\|_2^2$.

In your linear algebra class, you would have seen that the least squares solution is

$$\underline{x}^* = (A^T A)^{-1} A^T \underline{b}.$$

1.1.2 Constrained least squares

In the previous problem, we could impose additional constraints on the optimization vector (maybe we know something about this, the physical characteristics for example). Suppose that the vector $\underline{x}$ lies within a ball of radius r (This could be a power constraint, for instance). Then,

$$\underline{x}^* = \arg \min_{\|\underline{x}\| \leqslant r} \|A\underline{x} - \underline{b}\|^2.$$

1.1.3 Power allocation in Gaussian channels

A classical problem in information theory/wireless communication is to allocate power across multiple subchannels so as to maximize the achievable rate of communication. Suppose that we have a total power constraint of P, and this can be split across k subchannels, with the ith subchannel having an effective noise variance of σ_i^2. Then, the achievable rate is given by

$$R(\underline{P}) = \sum_{i=1}^{k} \frac{1}{2} \log_2 \left(1 + \frac{P_i}{\sigma_i^2} \right).$$

The power allocation problem can therefore be posed as the following optimization problem:

$$\underline{P}^* = \arg \max_{\underline{P}: \sum_i P_i \leqslant P} R(\underline{P})$$

1.1.4 Empirical risk minimization in machine learning

A common problem in machine learning is that of prediction. A lot of work on machine learning (deep neural networks, for example) can be abstractly thought of as curve fitting. Suppose that you want to solve a classification problem or a regression problem. In this case, we have the ground truth (for example, detecting whether a given image has a cat). We can think of this as an abstract function $g : \mathcal{I} \to \{0, 1\}$, where $\mathcal{I}$ is the set of all images, and g is a function whose value is 1 if the input image is a cat, and zero otherwise. Note that we *do not know* what this function is, or even a good model/approximation for this. However, this conceptually makes sense.

Neural networks can be thought of as a parametric class of functions, where the parameters are the "weights" (see wikipedia if you have not seen a neural network before). Every choice of the weights $\underline{w}$ defines a particular function $N^{(\underline{w})} : \mathcal{I} \to \{0, 1\}$. The learning problem is therefore one of finding that choice of weights that best approximates the true function g.

Unfortunately, we do not know what the true function is. However, we are given a *training set*, i.e., a collection of pairs $(I_1, t_1), (I_2, t_2), \ldots, (I_k, t_k)$, where $t_i = g(I_i)$ is the true value (whether there is a cat present in the image or not). To measure how well a given neural network approximates g on the training set, we use the following metric, called the empirical risk:

$$R(\underline{w}) = \frac{1}{k} \sum_{i=1}^{k} L(t_i, N^{(\underline{w})}(I_i)),$$

where L is a loss function. The problem of "training" a neural network is one of trying to solve the following empirical risk minimization problem:

$$\underline{w}^* = \arg\min_{\underline{w}} R(\underline{w}).$$

Unlike the problems mentioned previously, this problem is in general nonconvex. However, many techniques developed for solving convex optimization algorithms are used to design application-specific algorithms for training neural networks. See [1] to know more about empirical risk minimization.

1.1.5 Approximating maxcut in graphs and networks

Consider an undirected graph $(\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ denotes the set of vertices and $\mathcal{E}$ is the set of edges. If we take any subset of vertices $\mathcal{A} \subset \mathcal{V}$, and look at the edges going from $\mathcal{A}$ to $\mathcal{A}^c$, then this set of edges is called a cut. The *max cut* of the graph is the size of the maximum cut. Formally, the max cut problem can be stated as follows. Let A denote the adjacency matrix of the graph. Without loss of generality, let us assume that $\mathcal{V} = \{1, 2, \ldots, n\}$. Then, $a_{ij} = 1$ iff there is an edge between vertices i and j.

$$\text{MAXCUT} = \max_{\mathcal{A} \subset \mathcal{V}} \sum_{i \in \mathcal{V}} \sum_{j \in \mathcal{V}^c} a_{ij}$$

This is also equal to:

$$\text{MAXCUT} = \max_{\underline{x} \in \{\pm 1\}^n} \underline{x}^T L \underline{x}$$

where $L = D - A$ is called the graph Laplacian, and D is a diagonal matrix with the ith diagonal entry being equal to the degree of vertex i. Verify that the two optimization problems have the same solution.

In the above problem, the optimization variables take values in a finite set. This is an example of a *combinatorial optimization problem.* At first glance, it would therefore seem as though this should be an easier problem to solve than the the ones discussed previously. However, the problem is scaling this up. When n is large, the problem is computationally hard (in fact, this is an NP-hard problem).

One can therefore try to obtain approximate solutions. One approach is by considering a *relaxation*: we relax the constraints and/or the objective function to obtain a continuous-variable optimization that is easier to solve. One such relaxation is the so-called LP (linear programming) relaxation defined as follows:

$$\text{LP} - \text{MAXCUT} = \max_{X \in \mathcal{X}} \sum_{i,j} L_{ij} X_{ij}$$

where $\mathcal{X}$ is the set of all $n \times n$ matrices satisfying

$$\begin{gathered} -1 \leqslant X_{ij} \leqslant 1, \quad \forall i, j \\ X_{ii} = 1, \quad \forall i \\ X_{ij} + X_{jk} + X_{ik} \geqslant -1, \quad \forall i, j, k \\ X_{ij} - X_{jk} - X_{ik} \leqslant 1, \quad \forall i, j, k \end{gathered}$$

This is an example of a continuous-variable optimization problem where the objective function and all the constraints are linear. Such a problem is called a *linear programming* (LP) problem.

Except for specific types of graphs, the above approximation does not work well. The following semidefinite programming (SDP) relaxation is known to perform much better

$$\text{SDP} - \text{MAXCUT} = \max_{X \in \mathcal{X}_{SDP}} \sum_{i,j} L_{ij} X_{ij}$$

where $\mathcal{X}_{SDP}$ is the set of all *positive semidefinite matrices* satisfying $X_{ii} = 1$ for all i. This is an example of what is called a *semidefinite program*. Both relaxations above are specific subclasses of convex optimization problems.

See [2, Chapter 1] if you are interested to know more about this.

1.2 Convex optimization in ℝ

Let us quickly recap single variable convex optimization problems. This will give us the intution required to build the theory and analysis for multivariable problems.

1.2.1 Unconstrained optimization

Definition 1.2.1: Convex function of a single variable

A function $f : \mathbb{R} \to \mathbb{R}$ is said to be convex if for every $x_1, x_2 \in \mathbb{R}$ and every $0 \leqslant \alpha \leqslant 1$, we have

$$f(\alpha x_1 + (1 - \alpha)x_2) \leqslant \alpha f(x_1) + (1 - \alpha)f(x_2).$$

The function is said to be strictly convex if equality in the above holds only if $\alpha = 0$ or $\alpha = 1$.
A function f is (strictly) concave if $-f$ is (strictly) convex.

The geometric interpretation is the following: If you take any two points on the curve $y = f(x)$, then the right hand side is a point on the straight line joining $(x_1, f(x_1))$ and $(x_2, f(x_2))$. However, the left hand side is a point on the curve evaluated for some $x_1 \leqslant x \leqslant x_2$ (assuming that $x_1 < x_2$). The above statement says that f is convex iff the straight line joining two points on the curve always lies above the curve. See Fig. 1.1.

A very useful property is the following:

Lemma 1.2.1: Second derivative test

A function f for which $f''(x)$ exists everywhere is convex if and only if $f''(x) \geqslant 0$ for all $x \in \mathbb{R}$. It is strictly convex if $f''(x) > 0$ for all $x \in \mathbb{R}$.

To minimize a smooth convex function, we only need to compute the stationary point, i.e., the point at which $f'(x) = 0$.

Exercise: Prove that the stationary point is indeed the point of minimum.

1.2.2 Proof of Lemma 1.2.1

There are two statements to prove:

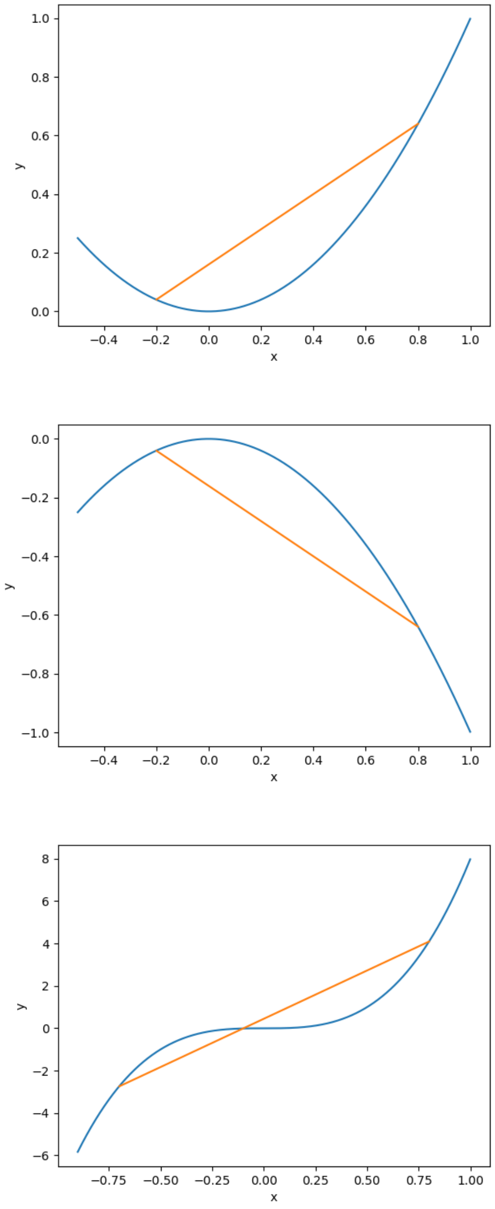

Figure 1.1: Plots of a convex function, a concave function, and a function that is neither convex nor concave.

- *If part*: If f'' exists and is nonnegative everywhere, then f is convex
- *Only if part*: If f is convex, and f'' exists everywhere, then $f"(x) \geqslant 0$ for all x.

Let us first prove the if part. We are given that $f''(x) \geqslant 0$ for all x. We will use the mean value theorem which says that if f is a continuous differentiable function, then for every $x_1 < x_2$, there exists a $x_1 < \beta < x_2$ such that

$$f'(\beta) = \frac{f(x_2) - f(x_1)}{x_2 - x_1}.$$

Consider any $x_1 < x_2$, and let $x = \alpha x_1 + (1 - \alpha x_2)$. We have,

$$\alpha f(x_1) + (1-\alpha) f(x_2) - f(x) = \alpha(f(x_1) - f(x)) + (1-\alpha)(f(x_2) - f(x))$$

Let us now use the mean value theorem for each of the two terms. There exist β_1, β_2 with $x_1 < \beta_1 < x < \beta_2 < x_2$ such that

$$\alpha f(x_1) + (1-\alpha) f(x_2) - f(x) = \alpha(x - x_1) f'(\beta_1) + (1-\alpha)(x_2 - x) f'(\beta_2).$$

Substitute for x in the above.

$$\alpha f(x_1) + (1-\alpha) f(x_2) - f(x) = \alpha(1-\alpha)(x_2 - x_1)(-f'(\beta_1)) + \alpha(1-\alpha)(x_2 - x_1) f'(\beta_2).$$

Since $f''(x) \geqslant 0$, we have $f'(\beta_2) \geqslant f'(\beta_1)$ for $\beta_2 > \beta_1$, and hence the right hand side is always nonnegative.

Let us now prove the only if part. Recall that

$$f''(x) = \lim_{t \downarrow 0} \frac{f(x+t) + f(x-t) - 2f(x)}{t^2}.$$

It is enough to show that $f(x+t) + f(x-t) - 2f(x) \geqslant 0$ for all $t > 0$.

Since f is convex,

$$f(x) = f\left(\frac{x+t}{2} + \frac{x-t}{2}\right) \leqslant \frac{1}{2} f(x+t) + \frac{1}{2} f(x-t).$$

which implies that $f(x+t) + f(x-t) \geqslant 2f(x)$, thus completing the proof.

You can redo the argument for strictly convex/concave functions. □

1.3 Numerically solving convex optimization problems

We will see three algorithms that let us obtain numerical solutions to one-dimensional convex optimization problems.

1.3.1 Golden section search

For this algorithm, we will assume that the function $f : \mathbb{R} \to \mathbb{R}$ is unimodal (i.e., has a unique local minimum and no maxima) in the interval $[a_0, b_0]$ over which we want to find the minimum. We do not require any other assumptions. This property is more general than convexity, and we do not even require the function to be differentiable for the algorithm to work.

Goal: Find $x^* = \arg \min_{x \in [a_0, b_0]} f(x)$

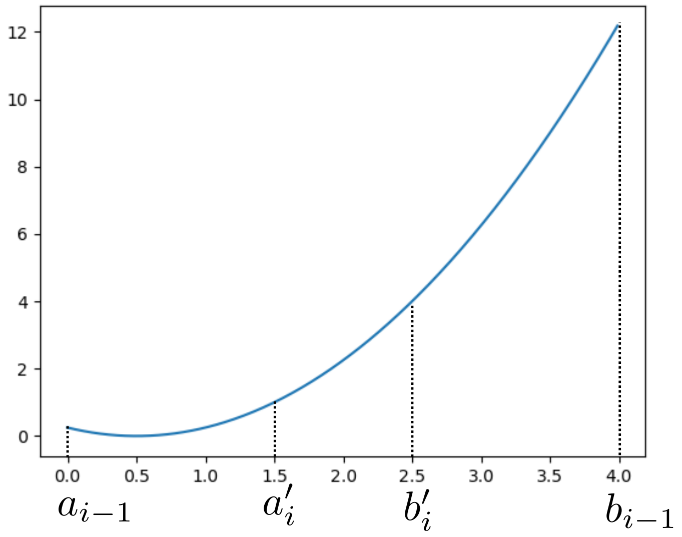

Figure 1.2: Illustrating the golden section search

up to an accuracy of $\epsilon > 0$.

Algorithm 1 describes the procedure. Initially, our search space for the minimum is the interval $[a_0, b_0]$. The algorithm iteratively shrinks the search space by a multiplicative factor of $(1 - \delta)$ per iteration.

Why does the algorithm work? See Fig. 1.2. Observe that if the minimum lies in $[a_{i-1}, a'_i]$, then $f(a'_i) \leqslant f(b'_i) \leqslant f(b_{i-1})$.

Show that if f is convex and $f(a'_i) \leqslant f(b'_i) \leqslant f(b_{i-1})$, then the minimum cannot lie in the interval (b_i, b_{i-1}). The argument is symmetric. Show that if $f(a_{i-1}) \geqslant f(a'_i) \geqslant f(b'_i)$, then the minimum cannot lie in $[a_{i-1}, a'_i]$.

We wish to choose δ so as to minimize the number of evaluations of f in order to reach the minimum. Clearly, we would need at most 2 evaluations of f (at a'_i and b'_i) per iteration. However, we can be clever, and try to choose δ so that in every step, $b'_{i+1} = a_i$ or $a'_{i+1} = b_i$. This would ensure that we only need to perform one new evaluation per iteration (except for the first iteration).

Solve for $b'_2 = a_1$, or equivalently,

$$b_1 - \delta(b_1 - a_0) = a_0 + \delta(b_0 - a_0)$$

We get

$$\delta = \frac{3 - \sqrt{5}}{2} \approx 0.382$$

and this partition is called the golden section.

After N steps, the range is reduced to $\approx (0.61803)^N (b_0 - a_0)$.

```
Given $f, a_0, b_0, \epsilon$;
Fix $\delta \in (0, 0.5)$;
$i \leftarrow 1$;
while $b_i - a_i > \epsilon$ do
    $a'_i \leftarrow a_{i-1} + \delta(b_{i-1} - a_{i-1})$;
    $b'_i \leftarrow b_{i-1} - \delta(b_{i-1} - a_{i-1})$;
    if $f(a_{i-1}) \geqslant f(a'_i) \geqslant f(b'_i)$ then
        $a_i \leftarrow a'_i$;
        $b_i \leftarrow b_{i-1}$;
    else
        $a_i \leftarrow a_{i-1}$;
        $b_i \leftarrow b'_i$;
    end
    $i \leftarrow i + 1$;
end
```

Algorithm 1: Search algorithm

1.3.2 Bisection method

If f is unimodal and the derivative exists in $[a_0, b_0]$, the we can obtain a better scheme. The basic principle is that $f'(x)$ is negative for x lying to the left of the stationary point, and is positive on the right side. The idea is to subdivide the interval into two equal parts. If the derivative at the midpoint is positive, then the minimum should lie on the left half. Otherwise, it would lie on the right half. See Algorithm 2.

In each step, the range is reduced by a factor of 1/2. Therefore, for a given $\epsilon > 0$, the bisection method will converge at least as fast as (but potentially faster than) the golden section search.

1.3.3 Newton's method

The previously described methods are essentially search algorithms that work well in one dimension. We now see an algorithm which can be easily extended to higher dimensional optimization problems.

The basic idea behind Newton's method is to approximate the function by a quadratic, and minimize the quadratic at each step. In each iteration t, we approximate f using the first three terms in the Taylor series expansion of f around x_{t-1}.

$$g_t(x) = f(x_{t-1}) + (x - x_{t-1})f'(x_{t-1}) + \frac{(x - x_{t-1})^2}{2} f''(x_{t-1}).$$

This is convex (why?) and can be minimized exactly. We take the minimum to be x_t. See Algorithm 3 and Fig. 1.3.

```
Given $f, a_0, b_0, \epsilon$;
$i \leftarrow 1$;
while $b_i - a_i > \epsilon$ do
    if $f'\left(\frac{b_{i-1}+a_{i-1}}{2}\right) > 0$ then
        $b_i \leftarrow \frac{a_{i-1}+b_{i-1}}{2}$;
        $a_i = a_{i-1}$;
    else
        if $f'\left(\frac{b_{i-1}+a_{i-1}}{2}\right) < 0$ then
            $a_i \leftarrow \frac{a_{i-1}+b_{i-1}}{2}$;
            $b_i = b_{i-1}$;
        else
            Output $\frac{a_{i-1}+b_{i-1}}{2}$;
            Terminate;
        end
    end
    $i \leftarrow i + 1$;
end
```

Algorithm 2: Bisection method

```
Given $f, x_0, \epsilon$;
$i \leftarrow 1$;
while $x_{i-1} - x_{i-2} > \epsilon$ do
    $x_i \leftarrow x_{i-1} - \frac{f'(x_{i-1})}{f''(x_{i-1})}$ $i \leftarrow i + 1$;
end
```

Algorithm 3: Newton's method

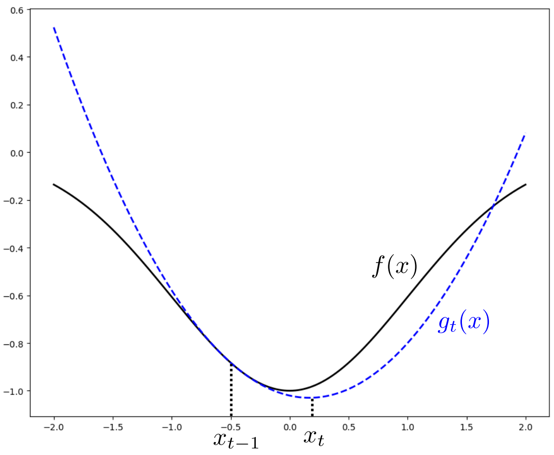

Figure 1.3: Illustrating one step of Newton's method

Chapter 2

Basics of Topology

This is a very brief introduction to basic concepts in topology. See [3, 4] or any other good book on real analysis/topology for more details.

2.1 Sets

2.1.1 Supremum, infimum, maximum and minimum

Given a nonempty set of real numbers $\mathcal{S} \subset \mathbb{R}$,

- A real $u \in \mathbb{R}$ is said to be an upper bound for $\mathcal{S}$ if $r \geqslant x$ for all $x \in \mathcal{S}$.
- A real $l \in \mathbb{R}$ is said to be a lower bound for $\mathcal{S}$ if $l \leqslant x$ for all $x \in \mathbb{S}$.
- u^* is said to be the supremum (also called the least upper bound) for $\mathcal{S}$ if u^* is an upper bound for $\mathcal{S}$ and if u is any other upper bound for $\mathcal{S}$, then $u \geqslant u^*$.
- l^* is said to be the infimum (also called the greatest lower bound) for $\mathcal{S}$ if l^* is a lower bound for $\mathcal{S}$ and if l is any other lower bound for $\mathcal{S}$, then $l \leqslant l^*$.
- If u^* is the supremum of $\mathcal{S}$ and $u^* \in \mathcal{S}$, then we say that u^* is the maximum of $\mathcal{S}$.
- If l^* is the infimum of $\mathcal{S}$ and $l^* \in \mathcal{S}$, then we say that l^* is the minimum of $\mathcal{S}$.

2.1.2 Functions

Consider any two sets $\mathcal{A}, \mathcal{B}$, and let $f : \mathcal{A} \to \mathcal{B}$ be a function. Then, we call $\mathcal{A}$ the domain and $\mathcal{B}$ the co-domain of f. The set $f(\mathcal{A}) \coloneqq \{y \in \mathcal{B} : y = f(x) \text{ for some } x \in \mathcal{A}\}$ is called the range of f. Note that the range could be a proper subset of $\mathcal{B}$.

For an arbitrary $\mathcal{S} \subset \mathcal{A}$, we call

$$f(\mathcal{S}) \coloneqq \{y \in \mathcal{B} : y = f(x) \text{ for some } x \in \mathcal{S}\}$$

the image of $\mathcal{S}$ under f. Similarly for $\mathcal{E} \subset \mathcal{B}$, we say that

$$f^{-1}(\mathcal{E}) \coloneqq \{x \in \mathcal{A} : f(x) \in \mathcal{E}\}$$

is the inverse image of $\mathcal{E}$ under f.

A function f is said to be injective (or one-to-one) if the inverse image $f^{-1}(\{y\})$ contains either zero or one element for every $y \in \mathcal{E}$. It is surjective (or onto) if the range is equal to the codomain. It is bijective if it is both injective and surjective.

2.1.3 How big is your set?

A set $\mathcal{A}$ is said to be finite if there exists a bijective map from $\mathcal{A}$ to $\{1, \ldots, n\}$ for some integer n. We call n, the cardinality of $\mathcal{A}$. If there exists no such map, then we say that $\mathcal{A}$ is infinite.

The set $\mathcal{A}$ is countable if there exists a injective map from $\mathcal{A}$ to the set of integers $\mathbb{Z}$. It is countably infinite if it is both countable and infinite. If $\mathcal{A}$ is infinite and there does not exists an injective map from $\mathcal{A}$ to $\mathbb{Z}$, then we say that $\mathcal{A}$ is uncountable (or uncountably infinite).

The following sets are countably infinite: $\mathbb{Z}, \mathbb{Z}^n, \mathbb{Z}_{\geqslant 0}, \mathbb{Q}, \mathbb{Q}^n$. The following sets are uncountable: $(0,1), \mathbb{R}, \mathbb{C}, \mathbb{R}^n$.

2.1.4 Metric and norm

Let $\mathcal{X}$ be an arbitrary set and $d : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be a function. Then, d is said to be a distance measure or a metric if it satisfies the following properties:

- $d(x_1, x_2) \geqslant 0$ for all $x_1, x_2 \in \mathcal{X}$, and $d(x_1, x_2) = 0$ if and only if $x_1 = x_2$,
- $d(x_1, x_2) = d(x_2, x_1)$ for all $x_1, x_2 \in \mathcal{X}$,
- $d(x_1, x_2) + d(x_2, x_3) \geqslant d(x_1, x_3)$ for all $x_1, x_2, x_3 \in \mathcal{X}$.

The pair $(\mathcal{X}, d)$ is called a metric space.

Let $\mathcal{X}$ be a vector space. A function $f : \mathcal{X} \to \mathbb{R}$ is said to be a norm if

- $f(\underline{x} + \underline{y}) \leqslant f(\underline{x}) + f(\underline{y})$ for all $\underline{x}, \underline{y} \in \mathcal{X}$
- $f(a\underline{x}) = |a| f(\underline{x})$ for all $\underline{x} \in \mathcal{X}$ and all $a \in \mathbb{R}$
- $f(\underline{x}) = 0$ if and only if $\underline{x} = \underline{0}$.

Let f be a norm on a vector space. Is $f : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ defined by $f(\underline{x} - \underline{y})$ a valid metric?

2.1.5 Neighborhood, closed and open sets

Neighborhood: Let $\mathcal{X}, d$ be a metric space. For any $r > 0$ and $x \in \mathcal{X}$, the set

$$\mathcal{N}_r(x) \coloneqq \{y \in \mathcal{X} : d(x, y) < r\}$$

is called a neighborhood (sometimes called open neighborhood) of x of radius r.

In $\mathbb{R}$, every neighborhood is of the form $(x - r, x + r)$. If we take the metric to be the standard euclidean distance, then the open neighborhood is simply an open ball of radius r.

Interior point: A point $x \in \mathcal{A}$ is an interior point of $\mathcal{A}$ if we can find an open neighborhood around x that is contained within $\mathcal{A}$. In other words, $\exists \epsilon > 0$ such that $\mathcal{N}_\epsilon(x) \subset \mathcal{A}$.

Open set: A set $\mathcal{A}$ is said to be an open set if for every $x \in \mathcal{A}$, we can find an open neighborhood of x that is contained within $\mathcal{A}$. In other words, for every $x \in \mathcal{A}$, there exists an $\epsilon > 0$ such that $\mathcal{N}_\epsilon(x) \subset \mathcal{A}$. Therefore, every point of $\mathcal{A}$ is an interior point. The interior of a set $\mathcal{A}$, denoted $\operatorname{int}(\mathcal{A})$ is the set of all interior points of $\mathcal{A}$.

Prove that (a, b) is open for every pair of real numbers $a < b$. Similarly, show that $[a, b]$ is not, by proving that you cannot find open neighborhoods at a, b.

Limit point: A point $x \in \mathcal{A}$ is a limit point of $\mathcal{A}$ if every open neighborhood of x contains at least one point from $\mathcal{A}$. The set of all limit points of $\mathcal{A}$ is called the closure of $\mathcal{A}$, and is denoted $\operatorname{cl}(\mathcal{A})$.

The boundary of $\mathcal{A}$ is

$$\operatorname{bd}(\mathcal{A}) \coloneqq \operatorname{cl}(\mathcal{A}) \backslash \operatorname{int}(\mathcal{A}).$$

Examples: For example, take $\mathcal{A} = [a, b]$. In this case, every point in $\mathcal{A}$ is a limit point because if you take any neighborhood $(x - \epsilon, x + \epsilon)$, then this has a nonempty intersection with $\mathcal{A} \backslash \{x\}$.

Similarly, if $\mathcal{A} = (0, 1)$, then every point of $\mathcal{A}$ is a limit point of $\mathcal{A}$. However, if we take the point a (or b), then this is also a limit point of $\mathcal{A}$ that lies outside a.

A more nontrivial example is the following: take $\mathcal{X} = \{1, 2, 3, 4, 5\}$. In this case, none of the points is a limit point. If we take the neighborhood $(0.5, 1.5)$ (neighborhood of radius 0.5 around 1), then this point only includes 1, and therefore $\mathcal{N}_{0.5}(1) \cap (\mathcal{X} \backslash \{1\})$ is empty. In fact, no point of $\mathbb{R}$ is a limit point of $\mathcal{X}$. To see why, take any $x \in \mathbb{R}$. If $x \notin \mathcal{X}$, then there is some real number $r > 0$ such that $\min_{y \in \mathcal{X}} |x - y| \geqslant r$. Then, $(x - r/2, x + r/2) \cap \mathcal{X}$ is empty and therefore there exists a neighborhood of x that does not contain $\mathcal{X}$.

Take the set $\{0.5^n : n = 1, 2, \ldots\}$. This set has one limit point at 0 (prove this).

Closed set: A set $\mathcal{X}$ is said to be closed if every limit point of $\mathcal{X}$ is a point of $\mathcal{X}$. In other words, the complement $\mathcal{X}^c$ should not contain any limit points of $\mathcal{X}$.

An equivalent but easier definition is that $\mathcal{X}$ is closed iff the limit of every convergent sequence of points from $\mathcal{X}$ also lies in $\mathcal{X}$.

Examples: Take $\mathcal{X} = [a, b]$. This is closed because every point that is not in $[a, b]$ cannot be a limit point of $\mathcal{X}$.

Take $\mathcal{X} = (a, b)$. In this case, a, b are also limit points of $\mathcal{X}$ (as discussed previously). But these do not belong to $\mathcal{X}$. Hence, $\mathcal{X}$ is not closed.

Take $\mathcal{X} = \{1, 2, 3, 4, 5\}$. For this set, no point in $\mathbb{R}$ is a limit point. Therefore, this set is closed.

Note

These definitions are rather formal, but several results in analysis crucially depend on them.

For any finite real numbers $a < b$, the set (a, b) is open, $[a, b]$ is closed, and $[a, b)$ is neither closed nor open. The set $\mathbb{R}$ is both open and closed!

One can also prove that the set $\mathcal{X}$ is closed if and only if the complement is open. Similarly, $\mathcal{X}$ is open if and only if $\mathcal{X}^c$ is closed.

2.1.6 Compactness

Compact set: A set $\mathcal{A}$ is compact if for every collection of open sets[1] $\{\mathcal{E}_\alpha : \alpha \in \mathcal{I}\}$ such that $\mathcal{A} \subset \bigcup_{\alpha \in \mathcal{I}} \mathcal{E}_\alpha$, we can find a finite subset of those, say, $\mathcal{E}_1, \mathcal{E}_2, \ldots, \mathcal{E}_n$ such that $\mathcal{A} \subset \bigcup_{i=1}^n \mathcal{E}_i$. Crucially for us, **every closed and bounded set in $\mathbb{R}^n$ is compact**.

Compactness is a very useful property: In general, compact subsets of metric spaces are closed, and closed subsets of compact sets are also compact.

2.1.7 Sequences and limits

A sequence in $\mathcal{X}$ is a countably infinite collection of elements from $\mathcal{X}$. Formally, a sequence can be thought of as a map from $\{1, 2, \ldots\}$ to $\mathcal{X}$. We will denote the sequence $a_1, a_2, \ldots$ by (a_n).

For example, $0.5, 0.5^2, 0.5^3, \ldots$ is a sequence in $\mathbb{Q}$ (also $\mathbb{R}$).

Let (a_n) be a sequence in a metric space. We say that a is a limit of the sequence if for every $\epsilon > 0$, we can find $N \in \mathbb{Z}_{>0}$ such that

$$d(a_n, a) \leqslant \epsilon, \text{ for all } n \geqslant N.$$

For example (a_n) with $a_n = 0.5^n$ has a limit (equal to 0). However, $a_n = 2^n$ does not have a real limit. The sequence $(1 + 1/n)^n$ has a limit in $\mathbb{R}$, but not in $\mathbb{Q}$. The sequence $(-1)^n$ does not have a limit at all.

2.1.8 Functions and limits

Let $\mathcal{X}, \mathcal{Y}$ be metric spaces and $f : \mathcal{X} \to \mathcal{Y}$. Then, we say that

$$\lim_{x \to p} f(x) = q$$

if for every $\epsilon > 0$, we can find $\delta > 0$ such that

$$d_{\mathcal{Y}}(f(x), q) < \epsilon$$

for all x such that

$$d_{\mathcal{X}}(x, p) < \delta.$$

A function $f : \mathcal{X} \to \mathcal{Y}$ is continuous if for every $p \in \mathcal{X}$, we have

$$\lim_{x \to p} f(x) = f(p).$$

It is not too hard to show that if $f : \mathcal{X} \to \mathcal{Y}$ is continuous, and $g : \mathcal{Y} \to \mathcal{Z}$ is continuous, then $h : \mathcal{X} \to \mathcal{Z}$ defined by $h(x) = g(f(x))$ is also continuous.

Lemma 2.1.1: Continuity and compactness

Let f be a continuous function from a compact set $\mathcal{X}$ to $\mathbb{R}^n$. Then, $f(\mathcal{X})$ is a closed and bounded set, and hence compact.
Moreover, if $f : \mathcal{X} \to \mathbb{R}$ where $\mathcal{X}$ is compact, then $f(\mathcal{X})$ has a maximum and a minimum.

[1] Such a collection of sets is called a cover for $\mathcal{A}$.

Note that the points of maximum/minimum need not be unique.

We can easily construct examples where violation of one of the properties can lead to the non-existence of a maximum/minimum. For example, if $\mathcal{X} = (0, 1)$ (noncompact), and $f(x) = x$, then this does not have a maximum or a minimum. Similarly, if f is not continuous, then this can lead to a problem. Take $\mathcal{X} = [0, 1]$, and let

$$f(x) = \begin{cases} x & \text{if } x > 0 \\ 1 & \text{if } x = 0 \end{cases}$$

is not continuous and does not have a minimum. The infimum is equal to zero, but this is never attained.

The above statements **do not imply** that continuity and compactness are **necessary** for the function to have a maximum and/or minimum. There are non-continuous functions defined over noncompact sets that have maxima and minima. These conditions are merely **sufficient** for the existence of maxima and minima.

2.1.9 Derivatives and gradients

Let $f : [a, b] \to \mathbb{R}$. The derivative of f at $x \in [a, b]$ is

$$f'(x) = \lim_{t \to 0} \frac{f(x+t) - f(x)}{t},$$

provided that the limit exists.

How do we generalize this to higher dimensions?

Definition 2.1.1: Gradient

Suppose that we have a function $f : \mathbb{R}^n \to \mathbb{R}^m$. Then, we say that f is differentiable at $\underline{x} \in \mathbb{R}^n$ if:

$$\lim_{\underline{z} \to \underline{x}} \frac{\|(f(\underline{x}) - f(\underline{z})) - D_{f(\underline{x})}(\underline{x} - \underline{z})\|}{\|\underline{x} - \underline{z}\|} = 0$$

for some $m \times n$ matrix $D_{f(\underline{x})}$. This is to indicate that the matrix depends on f and $\underline{x}$. Here, $\|\cdot\|$ denotes the ℓ_2 norm.
The matrix $D_{f(\underline{x})}$ is called the derivative (more commonly called the Jacobian) of f at $\underline{x}$.

Functions of one variable

For $f : \mathbb{R} \to \mathbb{R}$, the definition reduces to

$$\lim_{z \to x} \frac{|f(x) - f(z) - f'(x)(x - z)|}{|z - x|} = 0.$$

If $z > x$ and we take the limit $z \downarrow x$, then,

$$\lim_{z \downarrow x} \left| \frac{f(x) - f(z)}{z - x} - f'(x) \right| = 0,$$

which gives us the standard notion of derivative.

Real functions in $\mathbb{R}^n$

We want to show that for real-valued functions with vector arguments, the derivative in Definition 2.1.1 is equal to the transpose of the gradient that you have seen in your vector calculus course.

Let us suppose that $f : \mathbb{R}^n \to \mathbb{R}$. The gradient is some $D^T = \underline{u}$ (which is an $n \times 1$) vector that satisfies

$$\lim_{\underline{z}\to\underline{x}} \frac{\|(f(\underline{z}) - f(\underline{x})) - (\nabla f(x))^T(\underline{z} - \underline{x})\|}{\|\underline{z} - \underline{x}\|}$$

assuming that this exists and is unique.

Pick an arbitrary unit vector $\underline{v}$, and take $\underline{z} = f(\underline{x} + t\underline{v})$ and let $t \downarrow 0$. Then, we want

$$\lim_{t\downarrow 0} \frac{\|(f(\underline{x} + t\underline{v}) - f(\underline{x})) - \sum_{i=1}^n u_i(\delta v_i)\|}{\delta\|\underline{v}\|}$$

and this should hold for every unit vector $\underline{v}$. In other words

$$\lim_{t\downarrow 0} \left| \frac{f(\underline{x} + t\underline{v}) - f(\underline{x})}{t} - \sum_{i=1}^n u_i v_i \right|,$$

for every unit vector $\underline{v}$. Using the chain rule,

$$\lim_{t\downarrow 0} \frac{f(\underline{x} + t\underline{v}) - f(\underline{x})}{t} = \frac{df}{dt} = \sum_{i=1}^n \frac{\partial f}{\partial x_i}\frac{\partial x_i}{\partial t} = \sum_{i=1}^n \frac{\partial f(\underline{x})}{\partial x_i} v_i$$

Therefore,

$$\underline{u} = \nabla f(\underline{x}) = \left[\frac{\partial f}{\partial x_1}(\underline{x}), \dots, \frac{\partial f}{\partial x_n}(\underline{x})\right]^T.$$

Note: It is important to note that ∇f is a function from $\mathbb{R}^n \to \mathbb{R}^n$, which can be evaluated at some $\underline{x}$. A more appropriate notation would be $(\nabla f)(\underline{x})$, but this is more cumbersome to write.

For a general $f : \mathbb{R}^n \to \mathbb{R}^m$, a similar approach can be used to show that

$$(D_{f(\underline{x})})_{i,j} = \frac{\partial f_i}{\partial x_j}(\underline{x}).$$

Examples

Consider

$$f(\underline{x}) = \underline{x}^T A\underline{x} + \underline{b}^T\underline{x} + c,$$

where A is an $n \times n$ matrix, $b \in \mathbb{R}^n$, and $c \in \mathbb{R}$. Show that

$$\nabla f(\underline{x}) = (A + A^T)\underline{x} + \underline{b}$$

If A is symmetric, then the gradient is $2A\underline{x} + \underline{b}$.

At times, we will be interested in more complicated functions. Let $f : \mathbb{S}^n_{++} \to \mathbb{R}$, where $\mathbb{S}^n_{++}$ is the set of all $n \times n$ symmetric positive definite matrices, defined as

$$f(X) = \log\det X.$$

Basic calculus will not be enough to find the gradient in such problems. As we will see later, every symmetric positive definite matrix has a square root: an invertible matrix $X^{1/2}$ satisfying

$X = X^{1/2}X^{1/2}$. Let us denote the inverse of $X^{1/2}$ by X^{-1}. We will also make use of the property that if we can write

$$f(Z) = f(X) + \langle U, Z - X\rangle + g(Z - X),$$

where $g(Z - X) \to 0$ as $Z \to X$, then U is the gradient of f.

Using this, let us take $Z = X + \Delta X$, so that

$$\log\det Z = \log\det(X + \delta X) = \log\det(X^{1/2}(I + X^{-1/2}\Delta X X^{-1/2})X^{1/2})$$

which is

$$\log\det Z = \log\det X + \log\det(I + X^{-1/2}\Delta X X^{-1/2}).$$

If $\lambda_1, \dots, \lambda_n$ are the eigenvalues of $X^{-1/2}\Delta X X^{1/2}$, then

$$\log\det Z = \log\det X + \sum_{i=1}^{n} \log(1 + \lambda_i) \approx \log\det X + \sum_{i=1}^{n} \lambda_i$$

since $\log(1 + \lambda_i) \approx \lambda_i$ when $\lambda_i \approx 0$ (which is the case since $\Delta X \approx 0$). But the last term is simply the trace of $X^{-1/2}\Delta X^{-1/2}$, or the trace of $X^{-1}\Delta X$ — the inner product between X^{-1} and $(Z - X)$. Therefore,

$$\nabla f(X) = X^{-1}.$$

Exercise: Show that the *standard* inner product between two matrices defined as $\langle A, B\rangle := \sum_{i,j} a_{ij}b_{ij}$ is equal to the trace $\mathrm{tr}(A^T B)$.[2]

2.1.10 Chain rule for gradients

Let $f : \mathbb{R}^n \to \mathbb{R}^m$ and $g : \mathbb{R}^m \to \mathbb{R}^l$. Define $h : \mathbb{R}^n \to \mathbb{R}^l$ as $h(\underline{x}) = g(f(\underline{x}))$. If both f and g are differentiable, then so is h, and

$$D_{h(\underline{x})} = D_{g(f(\underline{x}))} D_{f(\underline{x})}.$$

In the special case where $f : \mathbb{R}^n \to \mathbb{R}$ and $g : \mathbb{R} \to \mathbb{R}$, we get

$$\nabla h(\underline{x}) = g'(f(\underline{x}))\nabla f(\underline{x}).$$

Example

If $g(\underline{x}) = f(A\underline{x} + \underline{b})$, then

$$\nabla g(\underline{x}) = A^T(\nabla f)(A\underline{x} + \underline{b}).$$

Similarly, find the gradients for:

- $f : \mathbb{R}^n \to \mathbb{R}$

$$f(\underline{x}) = \log\left(\sum_{i=1}^{m} e^{\underline{a}_i^T \underline{x} + b_i}\right)$$

- Fix symmetric positive definite $F_1, \dots, F_n$, and let domain of $\underline{x}$ be $\mathcal{X} = \{\underline{x} \in \mathbb{R}^n : \sum_i x_i F_i \text{ is symmetric PD}\}$.

$$f(\underline{x}) = \log\det(\sum_{i=1}^{n} x_i F_i)$$

[2] Note that in $\mathbb{R}^n$, $\langle \underline{x}, \underline{y}\rangle := \underline{x}^T A\underline{y}$ for any positive definite A is a valid inner product. Prove this. Similarly, more general inner products for matrices can be defined.

2.2 Second derivative

Recall that the second derivative of $f : \mathbb{R} \to \mathbb{R}$ is defined as the limit

$$f''(x) = \lim_{t\to 0} \frac{f'(x+t) - f'(x)}{t} = \lim_{t\to 0} \frac{f(x+t) + f(x-t) - 2f(x)}{t^2}$$

assuming that the limit above, and the first derivative exist.

We can define a similar quantity for vector-valued functions. The second derivative is essentially the gradient of the gradient.

Definition 2.2.1: Hessian

For $f : \mathbb{R}^n \to \mathbb{R}$, the Hessian matrix is an $n \times n$ matrix $\nabla^2 f$, whose (i, j)'th entry is

$$(\nabla^2 f(\underline{x}))_{i,j} = \frac{\partial^2 f}{\partial x_i \partial x_j}(\underline{x}),$$

provided that all the above partial derivatives exist.

For any $\underline{z}$ close to $\underline{x}$,

$$f(\underline{z}) = f(\underline{x}) + (\nabla f(\underline{x}))^T(\underline{z} - \underline{x}) + \frac{1}{2}(\underline{z} - \underline{x})^T(\nabla^2 f(\underline{x}))(\underline{z} - \underline{x}) + e(\underline{z}, \underline{x}),$$

where $e(\underline{z}, \underline{x})$ is a function that satisfies

$$\lim_{\underline{z}\to\underline{x}} \frac{e(\underline{z}, \underline{x})}{\|\underline{z} - \underline{x}\|^2} = 0.$$

Examples

Consider $f(\underline{x}) = \underline{x}^T A\underline{x} + \underline{b}^T\underline{x}$. Compute the Hessian.

2.2.1 Chain rule for second derivative

If $h(\underline{x}) = g(f(\underline{x}))$ for some $g : \mathbb{R} \to \mathbb{R}$ and $f : \mathbb{R}^n \to \mathbb{R}$, and both functions have first and second derivatives, then

$$\nabla^2 h(\underline{x}) = g''(f(\underline{x}))\nabla f(\underline{x})(\nabla f(\underline{x}))^T + g'(f(\underline{x}))\nabla^2 f(\underline{x}).$$

Chapter 3

Matrix Theory Fundamentals

This is a brief recap of concepts from matrix theory that we will require for this course. You are expected to have already gone through a course on linear algebra/matrices. See [5, 6] for a good introduction. Once you are comfortable with the basics (up to eigenvalues and eigenvectors), Horn and Johnson [7] is a good reference for more advanced topics.

3.1 Prerequisites

You should be familiar with the following general definition of a vector space.

Definition 3.1.1: Vector space

A *vector space* over $\mathbb{R}$ is a nonempty set $\mathcal{V}$ equipped with two operations: $+$ (vector addition) and $\cdot$ (scalar multiplication) that satisfies the following properties: for all $\underline{v}_1, \underline{v}_2, \underline{v}_3 \in \mathcal{V}$ and all $a, b \in \mathbb{R}$

- Commutativity of addition: $\underline{v}_1 + \underline{v}_2 = \underline{v}_2 + \underline{v}_1$
- Associativity of addition: $\underline{v}_1 + (\underline{v}_2 + \underline{v}_3) = (\underline{v}_1 + \underline{v}_2) + \underline{v}_3$
- Existence of zero vector (additive identity): there exists $\underline{0} \in \mathcal{V}$ such that $\underline{v}_1 + \underline{0} = \underline{v}1$.
- Existence of additive inverses: For every $\underline{v}_1 \in \mathcal{V}$, there exists $\underline{v}_m \in \mathcal{V}$ such that $\underline{v}_1 + \underline{v}_m = \underline{v}_m + \underline{v}_1 = \underline{0}$
- Associativity of multiplication: $(a \cdot b)\underline{v}_1 = a(b\underline{v}_1)$
- Unity preserves scaling: $1 \cdot \underline{v}_1 = \underline{v}_1$
- Distributivity: $a(\underline{v}_1 + \underline{v}_2) = a\underline{v}_1 + a\underline{v}_2$

We will mostly focus on finite-dimensional vector spaces over $\mathbb{R}$. However, vector spaces can be defined over more general fields (e.g., $\mathbb{C}, \mathbb{Q}$, finite fields, etc.).

3.1.1 Questions

You should also be be able to answer the following:

- What is a subspace of a vector space? Given a subset $\mathcal{S}$ of a vector space $\mathcal{V}$, do you need to test whether all 7 properties in Definition 3.1.1 are satisfied? Is there an easier test?
- How do you define linear combinations of vectors?
- What do you mean by linear independence?
- What do you mean by the span of vectors $\{\underline{v}_1, \underline{v}_2, \ldots, \underline{v}_n\}$?
- What is a spanning set of a vector space?
- When are vectors said to be linearly independent?
- What is a basis of a vector space? Is it unique?
- What is the dimension of a vector space?
- What are the four fundamental subspaces associated with a matrix?
- What is the rank of a matrix, and what is its nullity?
- How do you compute the rank or nullity of a given matrix? What is the computational complexity of doing so?
- What is the determinant of a matrix? How do you compute it, and what is the computational complexity of doing so?
- You should know what elementary row operations are, how to convert a matrix into the row reduced echelon form (RREF), and the QR decomposition of a matrix.
- Compute the rank, nullity, column space, and right null space of the following matrices

$$\begin{bmatrix} 1 & 3 \\ 2 & 6 \end{bmatrix}, \begin{bmatrix} 1 & 3 & 2 \\ 1 & 1 & 1 \\ 2 & 4 & 3 \end{bmatrix}, \begin{bmatrix} 1 & 3 & 2 \\ 3 & 1 & 1 \\ 2 & 1 & 3 \end{bmatrix}$$

3.1.2 Notation

We will typically use underlined lowercase letters to denote vectors, i.e., $\underline{u}, \underline{v}, \underline{w}$, etc. Unless otherwise mentioned, all vectors are column vectors. Uppercase letters will typically denote matrices (such as A, B, C). Boldface letters will be used to denote random variables, and underlined boldface letters will be used to denote random vectors (such as $\underline{\mathbf{x}}, \underline{\mathbf{y}}, \underline{\mathbf{z}}$).

3.2 Matrices and linear transformations

It is easy to see that every $m \times n$ matrix is a linear transformation from $\mathbb{R}^n$ to $\mathbb{R}^m$. A very important fact is that every linear transformation from $\mathbb{R}^n$ to $\mathbb{R}^m$ can be represented by an $m \times n$ matrix.

3.2.1 Determinants

Let $[n]$ denote the set $\{1, 2, \ldots, n\}$. A bijective map σ from $[n]$ to $[n]$ is called a *permutation*.

For example, $4, 2, 3, 1$ is a permutation of $[4]$, i.e., $\sigma(1) = 4, \sigma(2) = 2, \sigma(3) = 3, \sigma(4) = 1$.

What is the total number of possible permutations on $[n]$? A: $n!$.

Given any permutation σ of $[n]$, we can convert σ to $[n]$ by a sequence of pairwise exchanges of elements.

For example, $4, 2, 3, 1$ can be converted to $1, 2, 3, 4$ by a single pairwise exchange: exchange 1 and 4.

The permutation $4, 3, 2, 1$ can be converted to $1, 2, 3, 4$ using 2 pairwise exchanges:

$$(4, 3, 2, 1) \to (1, 3, 2, 4) \to (1, 2, 3, 4).$$

We could also do this using

$$(4, 3, 2, 1) \to (4, 3, 1, 2) \to (1, 3, 4, 2) \to (1, 2, 4, 3) \to (1, 2, 3, 4).$$

The number of pairwise exchanges to bring a permutation to $[n]$ is not unique. However, for a given σ, this is always odd or even. For example, if σ is $4, 3, 2, 1$, then the number of pairwise exchanges required to convert it to $1, 2, 3, 4$ is always even.

The *sign of a permutation* σ is defined to be $+1$ if the number of pairwise exchanges required to convert it to $[n]$ is even, and equal to -1 otherwise. This is denoted $\text{sgn}(\sigma)$.

Let $\mathcal{P}_n$ denote the set of all permutations over $[n]$.

Definition 3.2.1: Determinant

The determinant of an $n \times n$ matrix A is defined as

$$\det(A) = \sum_{\sigma \in \mathcal{P}_n} (-1)^{\text{sgn}(\sigma)} a_{1,\sigma(1)} a_{2,\sigma(2)} \cdots a_{n,\sigma(n)}.$$

Consider $n = 2$. There are two permutations of $[2]$, the identity permutation $(1, 2)$ and $(2, 1)$. The sign of $(1, 2)$ is $+1$ and that of $(2, 1)$ is -1. The determinant of a 2×2 matrix A is therefore equal to

$$\det(A) = a_{11}a_{22} - a_{12}a_{21}.$$

You can similarly compute the expression for the determinant of a 3×3 matrix, and verify that this indeed is the same as what you have seen in high school.

The *minor* of an element a_{ij} in a matrix A is the determinant of the submatrix obtained by deleting the ith row and jth column from A. The *cofactor* of a_{ij} is $(-1)^{i+j}$ times the minor of a_{ij}. Let us denote the cofactor as $\text{cof}_{ij}(A)$. We then have the following recursive definition of the determinant of a matrix (stated without proof):

$$\det(A) = \sum_{i=1}^{n} a_{ij}\text{cof}_{ij}(A) = \sum_{i=1}^{n} a_{ji}\text{cof}_{ji}(A)$$

for any $j \in [n]$.

Neither of the above definitions enable us to compute the determinant efficiently. The following is more useful:

- Any elementary row operation of type 1, i.e., multiplication all elements of a single row by a constant c, scales the determinant by c

  $$\det(B) = c\det(A)$$

  where B is obtained by multiplying all elements of (any) single row of A by c.

- Any elementary row operation of type 2, i.e., subtracting a row from any other row, does not change the determinant.

  $$\det(B) = \det(A)$$

  where the ith row of B is obtained by subtracting the jth row of A from the ith row of A (here $i \neq j$), and the remaining rows are unchanged.

- Any elementary row operation of type 3, i.e., exchanging two rows of A, scales the determinant by (-1).

  $$\det(B) = -\det(A)$$

  where B is obtained by exchanging two different rows of A.

Algorithm for computing the determinant

- Perform a sequence of row operations to reduce A to the row echelon form.
- Keep track of the constants $c_1, c_2, \ldots, c_l$ that are row multipliers in all the type-1 operations required.
- Count of number of type-3 operations required. Call this t_3.
- The determinant of the reduced matrix is 1. Therefore, $1 = (-1)^{t_3}(\prod_{j=1}^{l} c_j)\det(A)$, or

  $$\det(A) = \frac{(-1)^{t_3}}{\prod_{j=1}^{l} c_j}.$$

The determinant of an $n \times n$ matrix can therefore be obtained using $O(n^3)$ arithmetic operations.

3.2.2 Change of basis, similarity

Every $n \times n$ invertible matrix corresponds to a change of basis. If we want to represent a vector $\underline{v}$ (which is currently in the standard ordered basis) with respect to a new basis $\underline{b}_1, \ldots, \underline{b}_n$, then this is equivalent to finding the vector of coefficients $\underline{\alpha}$ such that

$$\underline{v} = \alpha_1 \underline{b}_1 + \cdots + \alpha_n \underline{b}_n$$

or

$$\underline{v} = P\underline{\alpha}$$

where $P = [\underline{b}_1, \ldots, \underline{b}_n]$.

Therefore, the vector $\underline{v}$ when represented in the new basis is

$$\underline{\alpha} = P^{-1}\underline{v}$$

If A, B are $n \times n$ matrices (not necessarily invertible), then we say that A is similar to B if there exists a change of basis matrix P such that

$$B = P^{-1}AP$$

Similar matrices can be viewed as the same linear transformation represented using a different basis. Effectively B is the same linear transformation as A, but instead with respect to the basis defined by the column vectors of P.

3.2.3 Gram-Schmidt orthogonalization

Recall that $\underline{v}_1, \underline{v}_2$ are orthonormal if both have unit norm and $\underline{v}_1^T \underline{v}_2 = 0$. A real matrix A is orthogonal if $A^{-1} = A^T$.

Given any vector space, we can construct an orthonormal basis using Gram-Schmidt orthogonalization. Start with any basis $\underline{v}_1, \ldots, \underline{v}_n$.

- Let $\underline{u}_1 = \underline{v}_1 / \|\underline{v}_1\|$.
- For $i = 2, 3, \ldots, n$:
  - Let
  $$\tilde{\underline{u}}_i = \underline{v}_i - \sum_{j=1}^{i-1} \langle \underline{u}_j, \underline{v}_i \rangle$$
  - Set
  $$\underline{u}_i = \frac{\tilde{\underline{u}}_i}{\|\tilde{\underline{u}}_i\|}.$$

Using the Gram-Schmidt process on the column vectors of any full rank matrix yields the QR decomposition of the matrix.

3.2.4 Eigenvalues and eigenvectors

Definition 3.2.2: Eigenvalues and Eigenvectors

Given an $n \times n$ real-valued matrix A, we say that $\lambda \in \mathbb{R}$ is an eigenvalue if there exists a nonzero $\underline{x} \in \mathbb{R}^n$ such that

$$A\underline{x} = \lambda \underline{x}.$$

The vector $\underline{x}$ is said to be an eigenvector corresponding to the eigenvalue λ.
The set of all eigenvalues of A is called the spectrum of A.

As you have seen in your linear algebra course, λ is an eigenvalue iff $(A - \lambda I)\underline{x} = 0$ has a nontrivial solution, implying that $A - \lambda I$ must be rank-deficient. This happens iff $\det(A - \lambda I) = 0$.

Lemma 3.2.1

The spectrum of A is the set of all solutions to the polynomial equation

$$\det(A - \lambda I) = 0.$$

The set of all eigenvectors of a given eigenvalue λ is the right nullspace of $A - \lambda I$. This is called the eigenspace corresponding to λ.

3.2.5 Exercises

Find the eigenvalues and corresponding eigenvectors for each of the following matrices:

- The all-ones matrix (i.e., the $n \times n$ matrix whose entries are all equal to 1)
- A general $n \times n$ diagonal matrix
- The all-zeros matrix
- The matrix
$$A = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \\ 1 & 3 & 4 \end{bmatrix}$$
- The matrix
$$A = \begin{bmatrix} 1 & 3 & 1 \\ 3 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Does every $n \times n$ matrix always have n real eigenvalues? Why or why not?

3.2.6 Diagonalizability

Definition 3.2.3: Diagonalizable matrix

An $n \times n$ real matrix A is said to be diagonalizable if it is similar to a diagonal matrix, i.e., there exists an $n \times n$ diagonal matrix D and an invertible matrix P such that

$$A = PDP^{-1}.$$

If λ is an eigenvalue of A, and k_a is the multiplicity of λ as a root of the characteristic polynomial $\det(A - \lambda I)$, then we say that k_a is the algebraic multiplicity of λ. The dimension of the right nullspace of $A - \lambda I$ is called the geometric multiplicity of λ.

Is the geometric multiplicity of an eigenvalue always equal to its algebraic multiplicity?

If the geometric multiplicity of every eigenvalue is equal to the algebraic multiplicity, then the matrix is diagonalizable. Note that the geometric multiplicity of every eigenvalue is at least 1 (why?). This implies that a matrix is (real) diagonalizable if all the eigenvalues are real and distinct.

It is easy to see that the set of all eigenvectors corresponding to an eigenvalue form a subspace. The dimension of this subspace (called the eigenspace) is equal to the geometric multiplicity.

The eigenvectors corresponding to distinct eigenvalues are linearly independent. Prove this. These properties imply that a matrix is diagonalizable if the geometric multiplicity is equal to the algebraic multiplicity.

A projection matrix P is an $n \times n$ matrix satisfying $P^2 = P$. If A is diagonalizable, then there exist projection matrices $P_1, \ldots, P_k$ (where k is the number of distinct eigenvalues of A) such that

$$A = \lambda_1 P_1 + \lambda_2 P_2 + \cdots + \lambda_k P_k$$

and

$$I = P_1 + P_2 + \cdots + P_k.$$

3.2.7 Symmetric and Positive Semidefinite Matrices

Suppose that A is symmetric. If λ_1, λ_2 are distinct eigenvalues and $\underline{v}_1, \underline{v}_2$ are corresponding eigenvectors, then

$$\underline{v}_1^T A \underline{v}_2 = \lambda_2 \underline{v}_1^T \underline{v}_2$$

However,

$$\lambda_2 \underline{v}_1^T \underline{v}_2 = (\lambda_2 \underline{v}_1^T \underline{v}_2)^T = (\underline{v}_1^T A \underline{v}_2)^T = \underline{v}_2^T A^T \underline{v}_1 = \underline{v}_2^T A \underline{v}_1 = \lambda_1 \underline{v}_1^T \underline{v}_2.$$

Since $\lambda_2 \neq \lambda_1$, the only possibility is that $\underline{v}_1^T \underline{v}_2 = 0$. Therefore, eigenvectors corresponding to distinct eigenvalues of a symmetric matrix are orthogonal. Since we can obtain an orthogonal basis for any subspace, the above statement implies that if a symmetric matrix is diagonalizable, then it is also orthogonally diagonalizable: There exists an orthogonal matrix P (with columns equal to eigenvectors) such that $A = PDP^T$.

Theorem 3.2.1: Spectral theorem

A real matrix A is symmetric if and only if it is orthogonally diagonalizable.

It is easy to show that if A is orthogonally diagonalizable, then it must be symmetric. If $A = PDP^T$ for orthogonal P and diagonal D. then $A^T = PD^T P^T = PDP^T = A$.

Here is a proof sketch to show that every symmetric matrix is orthogonally diagonalizable. Use induction. Any 1×1 matrix is orthogonally diagonalizable. Now, assume that the statement is true for all $(n-1) \times (n-1)$ matrices. For A, let λ_1 be an eigenvalue and $\underline{v}_1$ be a unit-norm eigenvector. We can extend this eigenvector to an orthonormal basis for $\mathbb{R}^n$. Let U be the matrix with these vectors as columns. Then,

$$A = P \begin{bmatrix} \lambda_1 & \underline{0}^T \\ \underline{0} & A' \end{bmatrix} P^T$$

for some matrix A'. Argue that this is true, and show that A' must also be symmetric. Now use the induction step to show that A' can be orthogonally diagonalized.

Definition 3.2.4: Positive Semidefinite matrices

A real symmetric matrix A is positive semidefinite (or nonnegative definite) if all its eigenvalues are nonnegative. We use the notation $A \succeq 0$ to denote that A is positive semidefinite (PSD), and $A \succeq B$ to denote that $A - B$ is PSD.
It is called positive definite (PD) if all the eigenvalues are positive. We say $A \succ B$ to indicate that $A - B$ is positive definite.
The set of all PSD matrices is denoted by $\mathbb{S}_+$, and the set of all PD matrices is denoted by $\mathbb{S}_{++}$.

For a positive semidefinite matrix, $\underline{x}^T A \underline{x} \geqslant 0$ for all $\underline{x} \in \mathbb{R}^n$. In fact, one can show that a matrix is positive semidefinite iff $\underline{x}^T A \underline{x} \geqslant 0$ for all $\underline{x} \in \mathbb{R}^n$.

The largest eigenvalue is equal to

$$\lambda_{\max} = \sup_{\underline{x} \neq \underline{0}} \frac{\underline{x}^T A \underline{x}}{\|\underline{x}\|^2}$$

and the smallest eigenvalue is equal to

$$\lambda_{\min} = \inf_{\underline{x} \neq \underline{0}} \frac{\underline{x}^T A \underline{x}}{\|\underline{x}\|^2}$$

To prove this, use the property that A is orthogonally diagonalizable, and express $\underline{x}$ in terms of the orthonormal basis. Then observe that $\underline{x}^T A \underline{x}$ is actually a convex combination of the eigenvalues and prove the above statements.

Square root: If A is positive semidefinite, then all the eigenvalues have square roots. If we define

$$A^{1/2} = P \begin{bmatrix} \sqrt{\lambda_1} & 0 & \cdots & 0 \\ 0 & \sqrt{\lambda_2} & & \\ \vdots & & \ddots & \\ 0 & & & \sqrt{\lambda_n} \end{bmatrix} P^T$$

then this matrix is also positive semidefinite and

$$A = A^{1/2} A^{1/2}.$$

The matrix $A^{1/2}$ is called the square root of A.

3.2.8 Singular Value Decomposition (SVD)

Any $m \times m$ matrix A with rank t can be written as

$$A = UDV^T$$

where U is a $m \times t$ matrix with orthonormal columns (called the left singular vectors), V is an $n \times t$ matrix with orthonormal columns (called the right singular vectors), and D is a $t \times t$ invertible diagonal matrix. The diagonal entries of D are called the singular values.

This can be obtained from the spectral theorem. The right singular vectors are essentially the (orthonormal) eigenvectors corresponding to nonzero eigenvalues of $A^T A$, the left singular vectors are the orthonormal eigenvectors corresponding tot he nonzero eigenvalues of AA^T, and the singular values are the square root of the nonzero eigenvalues of AA^T (or A^TA).

One can show that the largest singular value is

$$\sigma_{\max} = \sup_{\underline{x},\underline{y}\neq\underline{0}} \frac{\underline{x}^T A\underline{y}}{\|\underline{x}\|\|\underline{y}\|} = \sup_{\underline{x}\neq\underline{0}} \frac{\|A\underline{x}\|}{\|\underline{x}\|}.$$

The smallest singular value is defined to be zero if the matrix is not full-rank.

The condition number of an invertible $n \times n$ matrix A is

$$\operatorname{cond}(A) = \|A\|_2\|A^{-1}\|_2 = \frac{\sigma_{\max}}{\sigma_{\min}}.$$

It is said to be infinite if A is not invertible.

Chapter 4

Polyhedra and Linear Programs

4.1 Polyhedra

Recall that

$$\mathcal{H}(\underline{a}, b) \coloneqq \{\underline{x} \in \mathbb{R}^n : \underline{a}^T \underline{x} \leqslant b\}$$

defines a halfspace, and a polyhedron is an intersection of finitely many halfspaces.

4.2 Linear programming

An optimization problem of the following form is called a linear program

$$\min_{\underline{x} \in \mathcal{P}} \underline{a}^T \underline{x}$$

where $\mathcal{P}$ is is a polytope defined by a set of linear inequality and equality constraints:

$$\mathcal{P} = \{\underline{x} : \ \underline{a}_i^T \underline{x} \leqslant b_i, \ \underline{c}_j^T \underline{x} = d_j, \ 1 \leqslant i \leqslant m_1, \ 1 \leqslant j \leqslant m_2\}$$

A simple example is

$$\min_{\underline{x} : \substack{2x_1+3x_2+x_3 \leqslant 5 \\ x_2+x_3 \leqslant 4}} x_1 + x_2 + x_3$$

4.3 Maximum flow in a network

Consider a network consisting of n nodes, and the connectivity represented by a graph. Between each pair of nodes, say node i and node j, there is a bidirectional link between i and j. Out of these n nodes, we designate one of the nodes as a source (or sender) and a sink (or a receiver). The source wants to send information packets to the sink, and intermediate nodes are allowed to forward packets to neighbours. The goal is to design a forwarding protocol that allows the source to send the maximum number of packets per unit time to the sink.

The total maximum number of packets that can be exchanged between nodes i and j is limited to a number $c_{ij} = c_{ji}$, which is called the capacity of the link between i and j. For example, if $c_{ij} = 2$, then the number of packets sent from i to j plus the number of packets sent from j to i

per unit time should be at most 2 bits. We allow the number of packets per unit time sent from i to j to be fractional (not necessarily an integer).

Bibliography

[1] S. Shalev-Shwartz and S. Ben-David, *Understanding machine learning: From theory to algorithms.* Cambridge university press, 2014.

[2] D. Kunisky, “Lecture notes on sum-of-squares optimization,” 2022.

[3] W. Rudin *et al.*, *Principles of mathematical analysis*, vol. 3. McGraw-hill New York, 1976.

[4] T. Tao, *Analysis*, vol. 185. Springer, 2009.

[5] G. Strang, *Linear algebra and its applications.* Belmont, CA: Thomson, Brooks/Cole, 2006.

[6] J. Gilbert and L. Gilbert, *Linear algebra and matrix theory.* Elsevier, 2014.

[7] R. A. Horn and C. R. Johnson, *Matrix analysis.* Cambridge university press, 2012.