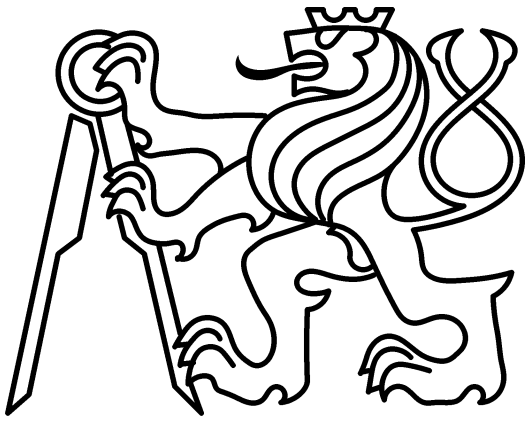
Czech Technical University in Prague

Faculty of Electrical Engineering

The Mathematics of Compressed Sensing in MRI

BACHELOR THESIS

Lenka Jacková
Medical Electronics and Bioinformatics
Prague, 2024



BACHELOR‘S THESIS ASSIGNMENT

I. Personal and study details

Student's name: **Jacková Lenka** Personal ID number: **507369**

Faculty / Institute: **Faculty of Electrical Engineering**

Department / Institute: **Department of Circuit Theory**

Study program: **Medical Electronics and Bioinformatics**

II. Bachelor’s thesis details

Bachelor’s thesis title in English:

The mathematics of compressed sensing in MRI

Bachelor’s thesis title in Czech:

Matematika komprimovaného snímání v MRI

Guidelines:

The method of compressed sensing is a mathematical method of signal reconstruction that allows for only few acquisitions of the measured signal. The method is modern: it originated in a series of papers in, roughly, 2006. The mathematics of the method is quite complex; it comprises ideas of advanced linear algebra, optimisation theory, probability theory and statistics.
The thesis will focus on only a subset of many issues that constitute the full method of compressed sensing. More in detail, the following topics will be covered by the student:
(1) The formulation of the basic problem of compressed sensing as a certain optimisation problem. Its computational complexity will be discussed.
(2) The understanding and explanation of particular two concepts that allow for a computationally tractable solution of the compressed sensing. Namely, the student will address the following:
(a) The Null Space Property (NSP) as a property that allows the hard problem of \ell_0 optimisation to be reduced to the significantly easier \ell_1 optimisation.
(b) The Restricted Isomorphism Property (RIP) is a property that (under mild additional assumptions) implies (NSP).
The points (1) and (2) above form only a tiny part of the complex tapestry of compressed sensing. The student will only hint at the ``full flavour'' of compressed sensing in a summarising non-technical chapter.
Although the main intended application of the topic is magnetic resonance imaging (MRI), the thesis will mention the physics and construction details of magnetic resonance only non-technically.
The thesis will be written obeying the usual mathematical standards. No new results will be proved in the thesis, the focus will be on the clarity of exposition of the ideas.

Bibliography / sources:

As main references the following list of publications will be used. However, of course, more references will be added in the course of the work on the thesis.
[1] Stephen Boyd and Lieven Vandenberghe, Convex optimization,
Cambridge University Press, 2004.
[2] Simon Foucart and Holger Rauhut, A mathematical introduction
to compressive sensing, Springer, 2013.
[3] Irina Rish and Genady Ya. Grabarnik, Sparse modeling,
CRC Press, 2015.

Name and workplace of bachelor's thesis supervisor:

doc. RNDr. Jiří Velebil, Ph.D. Department of Mathematics FEE

Name and workplace of second bachelor's thesis supervisor or consultant:

Date of bachelor's thesis assignment: **06.02.2024** Deadline for bachelor thesis submission: **24.05.2024**

Assignment valid until: **21.09.2025**

doc. RNDr. Jiří Velebil, Ph.D. Supervisor's signature	doc. Ing. Radoslav Bortel, Ph.D. Head of department's signature	prof. Mgr. Petr Páta, Ph.D. Dean's signature

III. Assignment receipt

The student acknowledges that the bachelor's thesis is an individual work. The student must produce her thesis without the assistance of others, with the exception of provided consultations. Within the bachelor's thesis, the author must state the names of consultants and include a list of references.

.	
Date of assignment receipt	Student's signature

Abstract

The Compressed Sensing method is a relatively new mathematical method that allows for signal reconstruction from a relatively small number of samples. Among other applications, it is also used in Magnetic Resonance Imaging (MRI). Compressed Sensing can reduce MR examination time, which is advantageous for patients because the examination is very uncomfortable for most of them. In this thesis we will delve into the mathematics of Compressed Sensing. Since this topic is very complex, emphasis will be put only on certain aspects. We will mainly address the problem of an ℓ_0 minimisation and its transformation into an ℓ_1 minimisation, the Restricted Isometry Property and the Null Space Property of matrices. Finally, we will discuss random matrices having Restricted Isometry Property.

keywords: compressed sensing, basis pursuit, null space property, restricted isometry property

Abstrakt

Metoda komprimovaného snímání je relativně nová matematická metoda, která umožňuje rekonstrukci signálu z poměrně malého množství vzorků. Mimo jiné se využívá také při zobrazování pomocí magnetické rezonance. Komprimované snímání umožňuje snížení času potřebného na vyšetření magnetickou rezonancí, což je výhodné pro pacienty, protože pro většinu je vyšetřování velmi nepříjemné. V této práci se budeme věnovat matematice komprimovaného snímání. Protože je toto téma velmi obsáhlé, budeme klást důraz pouze na několik aspektů. Budeme se zabývat problémem ℓ_0 minimalizace a jeho převedením na ℓ_1 minimalizaci, vlastností zeslabené izometrie a vlastností nulového prostoru matic. Na závěr pojednáme o skutečnosti, že náhodné matice mají vlastnost zeslabené izometrie.

klíčová slova: komprimované snímání, lineární program, vlastnost zeslabené izometrie, vlastnost nulového prostoru

Declaration

I declare that I elaborated this thesis on my own and that I mentioned all the information sources that have been used in accordance with the Guideline for adhering to ethical principles in the course of elaborating an academic final thesis.

In Prague, 23 May 2024 Lenka Jacková

Acknowledgements

I would like to express my deepest gratitude to the supervisor of the thesis Jiří Velebil for his patience, valuable advice and for creating a congenial atmosphere to work in. I am grateful for a big motivation, thanks to which I was able to complete the thesis without being under stress.
Then I would like to thank my family, especially my parents and my sister, for their support not only during my studies. Special thanks to my uncle for the proofreading and constructive feedback. I am also thankful to my friends and to Tomáš for standing by me.

Contents

Introduction **6**

1 A very brief introduction to magnetic resonance (MR) **8**
1.1 The history of MR . . . 9
1.2 Elementary physical principles of MR . . . 9
1.3 Health risks of MR . . . 10
1.4 Reconstructing the measured data . . . 11

2 The problem of magnetic resonance image (MRI) reconstruction **12**
2.1 The general idea of a very basic lossy compression . . . 12
2.2 Sampling: Making the continuous signal discrete . . . 13
2.3 Compressed Sensing: A very sketchy overview . . . 13

3 The complexity classes of algorithms **15**
3.1 Basic complexity classes: P, NP and NP-hard problems . . . 16
3.2 The complexity of optimisation problems . . . 18

4 An introduction to ℓ_p norms optimisation **20**
4.1 ℓ_p norm . . . 20
4.2 The matrix calculus . . . 22
4.3 ℓ_2 norm optimisation . . . 24
4.4 ℓ_0 norm optimisation . . . 26

5 Basis Pursuit **29**
5.1 ℓ_1 norm optimisation . . . 29
5.2 Basis Pursuit as a linear program . . . 31

6 The Null Space Property **33**
6.1 Nullifying components of a vector . . . 33
6.2 The Null Space Property and sparse recovery . . . 34

7 The Restricted Isometry Property **38**
7.1 Shrinking vectors and matrices . . . 38
7.2 Additional facts about norms . . . 39
7.3 The Restricted Isometry Property . . . 41

8 The Restricted Isometry Property for random matrices **46**
8.1 A quick overview of probability theory . . . 46
8.2 A brief insight into the Fourier transform . . . 48
8.3 The Restricted Isometry Property theorem for random matrices . . . 49

Summary **51**

Bibliography **53**

Introduction

This paper explores the mathematics of Compressed Sensing. Compressed Sensing is a relatively new method (introduced in 2006), and its applications in signal processing is extensive. One of them is in medicine, namely in Magnetic Resonance Imaging (MRI). MRI is an impressive method that allows the detection of various illnesses, thus preventing their spread in time. While not inherently dangerous for patients, the method confines patients to a narrow tunnel and exposes them to loud noises, which may prove to be extremely uncomfortable for some of them. However, thanks to the Compressed Sensing technique, the time required for an examination can be significantly reduced, which improves the patient's experience and, also, reduces the operational costs.

(1) In **Chapter** 1, we briefly examine the purpose of magnetic resonance (MR) and its associated risks. Additionally, we give insights into the history of MR and a short overview of how MR works from the physics perspective.

(2) **Chapter** 2 focuses on signal reconstruction, specifically image reconstruction. In the first part we talk about the classical (lossy) compression. We also mention sampling, introducing the famous Nyquist-Shannon Sampling Theorem. The last part explains the fundamental concept of Compressed Sensing and formulates the problem we will focus on in the rest of the thesis.

(3) The following **Chapter** 3 delves into algorithm theory. The first section provides a brief historical background, followed by, in the second section, an introduction of the basic complexity classes, including the necessary definitions.

(4) The introduction of **Chapter** 4 presents the so-called ℓ_p norms. In the following sections, we centre our attention on the ℓ_2 and ℓ_0 norms. The ℓ_2 norm is the well-known Euclidean norm, and we demonstrate that minimisation of the ℓ_2 norm is an easy problem. On the other hand, the ℓ_0 norm is not an honest norm, and we illustrate that its minimisation is inherently a hard problem. Unfortunately, ℓ_0 optimisation is an issue we need to solve in the concept of Compressed Sensing.

(5) The hard problem of ℓ_0 optimisation can be solved in several ways, and this thesis, namely **Chapter** 5, presents one of the methods: we transform the problem into an ℓ_1 optimisation problem and moreover, we show that ℓ_1 minimisation problem can be written as a linear program. We closely examine the conditions under which this transformation from ℓ_0 to ℓ_1 is possible in next chapters.

(6) **Chapter** 6 deals with a property of matrices called the Null Space Property. When it is satisfied it means that the transformation into the ℓ_1 optimisation problem is possible. This may sound promising but, unfortunately, verifying whether a matrix has the Null Space Property is hard.

(7) In **Chapter** 7 we introduce different solution which is the Restricted Isometry Property, another property of matrices. We show that having the Restricted Isometry Property can imply, under some conditions, having the Null Space Property.

(8) Finally, matrices that have the Restricted Isometry Property are discussed in **Chapter** 8, where we take a brief look into the problematics of random matrices.

All of the above results are well-known and references to the existing literature are provided. The import of the thesis is to give a readable text devoted only to some aspects of the theory of Compressed Sensing.

Chapter 1

A very brief introduction to magnetic resonance (MR)

Examination by means of a magnetic resonance scanner is a modern medical imaging technique offering an alternative to taking, say, X-ray images. The magnetic resonance scanner uses the impact of varying magnetic fields to the spin of hydrogen atoms in the human body. Since hydrogen atoms form around 10 % of a human body and since in important parts of a human body hydrogen is abundant (e.g., water itself, proteins, DNA), scanning hydrogen in a human body makes magnetic resonance a very versatile diagnostic technique.

Although magnetic resonance scanners do not use any ionising radiation, their use can have a negative effect on a patient. Perhaps the most visible effect is the fact that the patient is essentially put into a narrow tunnel, left alone in quite a noisy environment and, last but not least, is exposed to quite a strong magnetic field. While the examination may not be explicitly dangerous for humans, it may pose a significant problem for patients using devices such as cardiac pacemakers. More health risks are discussed in Section 1.3.

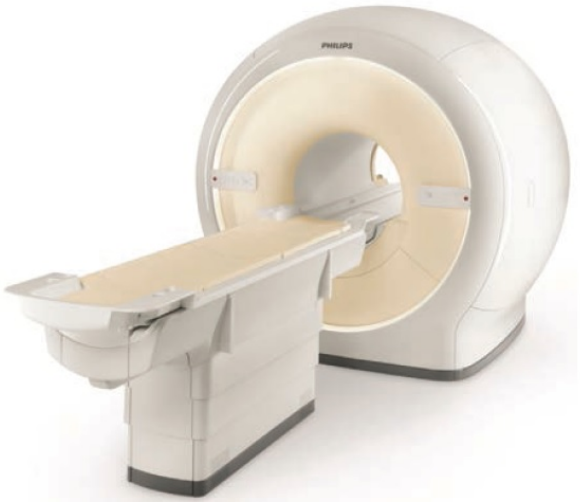


Figure 1.1: The MRI device, picture taken from [21].

In this brief chapter we discuss the above aspects of magnetic resonance and we argue why it is rather desirable to minimise the duration of the examination by a magnetic resonance scanner. We also indicate the problem of *Magnetic Resonance Imaging* (MRI), i.e., the reconstruction of measured data into the picture of examined tissue.

1.1 The history of MR

The history of magnetic resonance imaging dates back to the early 20th century and includes many significant scientists who contributed to the development of this important technology. One of the key figures was Isidor Isaac Rabi, who dedicated his research to the magnetic properties of atoms and was awarded the Nobel Prize for his work in 1944. Felix Bloch and Edward Purcell received Nobel Prize in 1952 for their advancements in nuclear magnetism and new methods of precision measurement. Raymon Vahan Damadian pioneered the application of MR for disease diagnosis, especially cancer, and his work in 1971 laid the groundwork for the medical use of this technology. Paul Lauterbur and Peter Mansfield, also prominent scientists, made key innovations in MR imaging in 1973 and 1977. Their work led to the modern MR as we know it today, and both were awarded the Nobel Prize in Physiology or Medicine for their significant contributions in medical imaging.

Figure 1.2: Isidor Isaac Rabi (1898 – 1988), picture taken from [37]

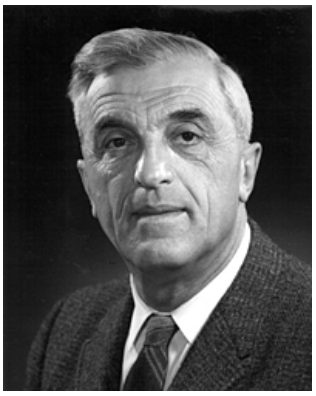


Figure 1.3: Felix Bloch (1905 – 1983), picture taken from [38]

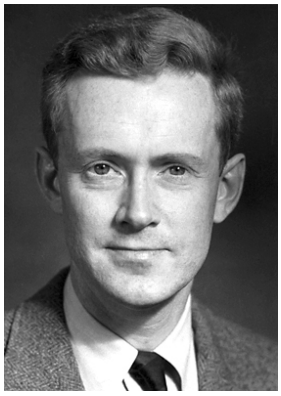


Figure 1.4: Edward Mills Purcell (1912 – 1997), picture taken from [39]

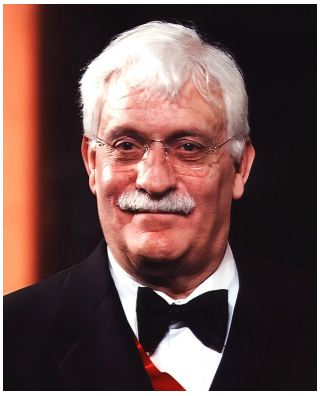


Figure 1.5: Raymond Vahan Damadian (1936 – 2022), picture taken from [25]

Figure 1.6: Paul Christian Lauterbur (1929 – 2007), picture taken from [40]

Figure 1.7: Peter Mansfield (1933 – 2017), picture taken from [41]

1.2 Elementary physical principles of MR

Now, let us briefly examine how MR works from a physics perspective. To begin with, let us state several fundamental facts. The human body is formed of atoms, which contain a nucleus with protons and neutrons. A proton has a positive electric charge and a spin. The flow of electric charge is an electrical current, and wherever there is an electric current, a magnetic field is generated. When protons are placed to an external magnetic field, they may align themselves either in parallel or anti-parallel to the field, and each alignment

requires a different amount of energy. In the context of MR, we prefer a lower energy level, which corresponds to the parallel alignment and is more frequent among protons.

When protons align in parallel and anti-parallel configurations, their magnetic forces cancel each other out. However, due to the prevalence of the parallel alignment, there are non-cancelled magnetic forces in the direction of the external magnetic field. That is known as longitudinal magnetisation. It is important to note that protons within the external magnetic field are not at rest, they are moving around; this movement is called precession. The velocity of this movement depends on the strength of the magnetic field and it is measured as the precession frequency.

When a patient (composed of an enormous amount of protons) is placed in an MR machine, due to a larger number of parallely-alligned protons, we can say that a patient becomes a magnet with his own magnetic field with direction the same as external field. Then a radio electromagnetic wave, in MR named a radio frequency (RF) pulse, is sent. This pulse is designed to disturb the alignment of protons and exchange energy with them. To achieve this state, the frequency of the pulse must match the precession frequency of the protons which is called resonance. Resonance results in absorbing energy from the radio wave by some protons, leading to a decrease in longitudinal magnetisation and to the establishment of transverse magnetisation. The transverse magnetic vector is moving with the precessing protons and induces an electric current, resulting in a measurable signal.

Once the RF pulse is switched off, a gradual return to the initial state occurs. The longitudinal magnetisation amplifies back to its original value and the transversal magnetisation completely vanishes in the end. These processes are happening independently. The increase in longitudinal magnetisation takes about 300 to 2000 ms, which is expressed by the constant called $\mathbf{T_1}$. The decrease in transversal magnetisation is 2–10 times shorter, lasting 30 to 150 ms, and it is expressed by the constant $\mathbf{T_2}$. The times $\mathbf{T_1}$ and $\mathbf{T_2}$ depend on the tissue structure and the intensity of the external magnetic field. In different tissues these processes are of varying durations. For example, $\mathbf{T_1}$ in water is much larger than $\mathbf{T_1}$ in fat. If we plot magnetisation against time after the RF pulse is switched off, we obtain curves called $\mathbf{T_1}$ curve and $\mathbf{T_2}$ curve, which can be seen in the images below.

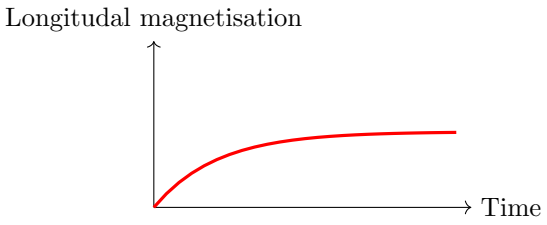


Figure 1.8: $\mathbf{T_1}$ curve

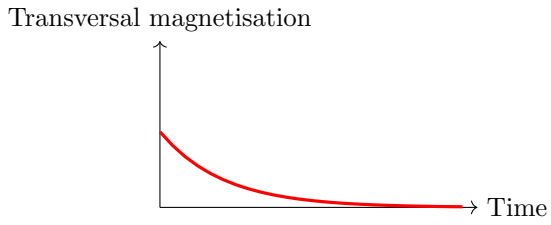


Figure 1.9: $\mathbf{T_2}$ curve

This is used in the so-called "pulse sequence", where multiple RF pulses are sequentially sent and individual tissues can be distinguished. Time between the transmitted pulses TR (time to repeat) is fundamental in shaping the resulting image. The direction of the RF pulse also has an impact, it is frequently utilising 90-degree RF pulses (those that deflect the magnetisation vector by 90 degrees) and 180-degree RF pulses.

Further information on the physics of MR
This paper is not concerned with the actual physics of MR nor with the medical technicalities of MR. We concentrate our efforts upon the mathematical problem of processing the scanned information. For the physics involved in MR we refer the reader to [36]. For the medical practicalities of MR we refer to [21].

1.3 Health risks of MR

So far, no negative side-effects of MR scans on the human body are known; however, there are some disadvantages of using it. MR has high acquisition and operational costs, and compared to other diagnostic procedures, it requires a longer examination time (30–60 minutes). The confined space of the MR scanner can be problematic for patients suffering from claustrophobia. Additionally, MR produces loud noise, so it is very important to use hearing protection. For patients with cardiac pacemakers, cochlear implants or other internal electrical devices, there is a potential risk, and consultation with doctors is necessary. The presence of metallic objects, such as artificial valves, and pregnancy also requires consideration of whether MR is safe to use.

1.4 Reconstructing the measured data

The measured data (and their subsequent recording) obtained during magnetic resonance examination are not designed to produce images directly. This places the reconstruction and/or imaging problem within the realm of various *reconstruction problems* known from Signal Theory.

To simplify, we may distinguish between two approaches to signal reconstruction.

(1) **Classical data compression.** The measured data are usually quite massive, and for successful reconstruction only a small part of these data is essential. Thus, one can *compress* the measured data and use the compressed data for reconstruction.

(2) **Compressed Sensing.** Since it is typically the case that lots of measured data can be "thrown away", one might ask whether a substantially smaller number of measurements for a successful reconstruction would be sufficient. Thus, one would *compress* the measurements themselves.

The approach (2) is quite recent and promising. In fact, this approach — when applied to magnetic resonance — would address the health risks described in Section 1.3 above.

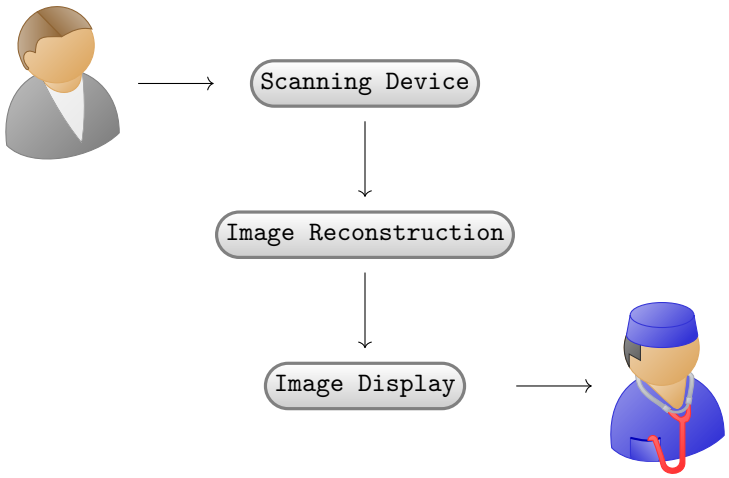
We compare approaches (1) and (2) in the next chapter of this thesis and we will focus on some aspects of (2) in the rest of this text.

Chapter 2

The problem of magnetic resonance image (MRI) reconstruction

The role of image reconstruction in medicine is to collect the signals from a medical device and produce an image meaningful to medical experts so that they can analyse it and suggest a diagnosis and/or therapy.

Thus, we obtain the following scheme:



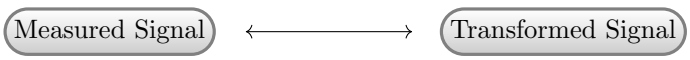
Although the method of *Compressed Sensing* that we focus on in this text is particularly suitable for MRI, in the current chapter we treat the scanning device as unspecified. Moreover, we intentionally speak rather informally in this chapter: we only indicate how lossy compression works, how signals are digitised and we sketch the basic idea of Compressed Sensing.

2.1 The general idea of a very basic lossy compression

A continuous signal from the scanning device is usually transformed into a different basis. For example, when using the *Fourier transform*,[1] we can convert the signal into a linear combination of sine and cosine functions. In this way we create a series that typically contains an infinite number of terms.

The transformation that decomposes the continuous signal in this way is bijective. Therefore, once we have this decomposition into sine and cosine functions available, we can use the inverse Fourier transform to precisely reconstruct the measured signal.

Thus, we have obtained a bijection



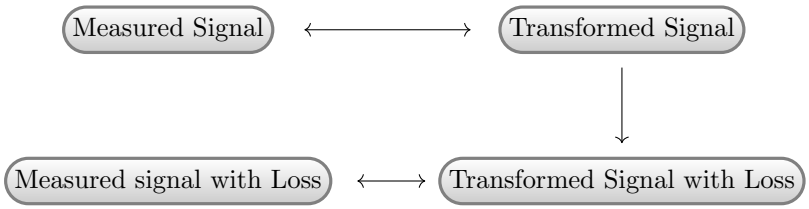
[1] Other types of transforms can be used, we stick with the Fourier transform just for simplicity.

stating that we can talk either about the measured signal or of its transform.

However, the measured data are usually unnecessarily large and they contain some information that may be irrelevant to us. This is due to the limitations of human perception, such as the human ear not hearing very high frequencies, which are not needed in a file for everyday music listening. Similarly, in images, the human eye is unable to recognise subtle transitions between colours or intensities. And now comes the turn of lossy compression. After applying the Fourier transform to the measured signal, we choose only certain terms that interest us. The resulting series is usually no longer infinite. Thus, we are dealing with finite sums, entering therefore the realm of finite dimensional problems. Although we lose some of the original data, this difference is often not noticeable to human senses.

If we choose our transformation cleverly (the choice is usually very case-specific, i.e., having knowledge of the physiology of a human ear, etc.), then which piece of the information we can safely "throw away" can be read from the transformed signal.

Therefore we obtain the following picture



and in the problem of lossy compression, the goal is not to distort the measured signal "too much".

2.2 Sampling: Making the continuous signal discrete

Another approach to lossy compression is to replace the continuous signal by a *discrete* signal at the very outset. This is usually done by some *Analog-Digital Converter*. To illustrate roughly how this works, we may imagine a sound wave as the input signal. By recording the sound only at certain times, we obtain a discrete signal that is usually modelled as a vector (with very many components).

But how do we know which samples are important and which ones we do not need? The *Nyquist–Shannon Sampling Theorem* [24], [29] provides an answer:

> The reconstruction of a continuous, frequency-limited signal from its samples is possible if the sampling frequency is higher than twice the highest harmonic component of the sampled signal.

For example, human hearing processes the range form 20 Hz to 20,000 Hz, thus, the sampling frequency for sound is usually 44,100 Hz.

2.3 Compressed Sensing: A very sketchy overview

As shown in the early 21st century, the sampling condition of the Nyquist-Shannon Sampling Theorem can be "bypassed", provided the signal is "very sparse". This reconstruction method is called *Compressed Sensing*.

The idea of Compressed Sensing is as follows: why do we need so many measurements when, in the end, we will not use a large amount of the data? In our specific case, this raises the question of why a patient has to be under an MR machine for such a long time. Would it not be possible to reconstruct the result from a smaller number of measurements? Emmanuel Candés, Justin Romberg, Terence Tao, and David Donoho demonstrated that it is possible in a series of papers around 2006. This process is not data compression but rather measurement compression.

To hint at how this works, we will now (informally) introduce the notation that we use throughout the text. Compressed Sensing in comprehensive form can be found in, e.g., [15], [27].

(1) We start with a measured signal $\mathbf{x}$ (represented as a vector with n components) and we choose a regular $n \times n$ matrix $\mathbf{\Psi}$.

Moreover, we denote by $\mathbf{s}$ the vector satisfying

$$\mathbf{x} = \mathbf{\Psi s}$$

Thus, in the language of Section 2.1, the matrix $\mathbf{\Psi}$ represents the discrete version of the (inverse Fourier) transform and the vector $\mathbf{s}$ represents the transformed signal.

(2) Observe that the vectors $\mathbf{x}$ and $\mathbf{s}$ are in bijective correspondence, since the matrix $\mathbf{\Psi}$ is assumed to be regular.

However, the nature of many processes (including MRI) is such that the vector $\mathbf{s}$ is *very sparse* (i.e., it contains mainly zeroes).

This indicates that one could design a *wide* matrix $\mathbf{C}$ (i.e., such that $\mathbf{C}$ has as many columns as $\mathbf{x}$ has components but $\mathbf{C}$ has *considerably fewer* rows than columns). By defining the vector $\mathbf{y}$ by the equality

$$\mathbf{y} = \mathbf{C}\mathbf{x}$$

we could try to solve the equation

$$\mathbf{y} = \underbrace{\mathbf{C}\mathbf{\Psi}}_{=\mathbf{\Theta}}\mathbf{s}$$

for $\mathbf{s}$ with the additional requirement that $\mathbf{s}$ is as sparse as possible.

What the equations look like and what dimensions the matrices have can be clearly seen in the following image.

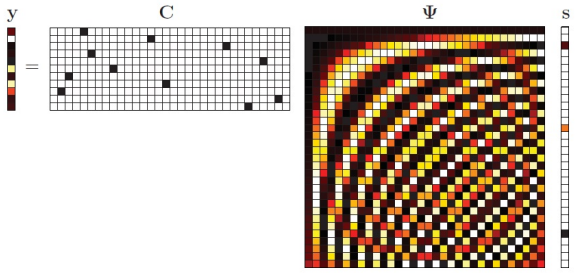


Figure 2.1: Scheme of measurements in the Compressed Sensing, picture taken from [5]

(3) Once an $\mathbf{s}$ as in (2) above is found, we can obtain the resulting $\mathbf{x}$ through (inverse Fourier) transform $\mathbf{\Psi}$.

Thus, the whole problem of Compressed Sensing can be formulated as an *optimisation problem*:

Given a "wide" matrix $\mathbf{\Theta}$ and a vector $\mathbf{y}$, we solve the equation $\mathbf{y} = \mathbf{\Theta}\mathbf{s}$ under the condition that $\mathbf{s}$ is as sparse as possible (i.e., the so-called ℓ_0 norm $\|\mathbf{s}\|_0$ of $\mathbf{s}$ is minimal).

As we will show in Section 4.4, this problem is *NP*-hard (the meaning of *NP*-hard will be explained in Chapter 3 that follows). Since we do not want to give up on the above idea of Compressed Sensing, we need to develop certain notions and techniques that will allow to trade the above inherently hard problem for a tractable one.

Chapter 3

The complexity classes of algorithms

Algorithms have existed in our world for a long time, we find them for example as cooking recipes or instructions for children's building blocks. But the question is how to formalise the concept of an algorithm. In a mathematical context, we can define an algorithm as follows:

> An algorithm is a process or a set of rules to be followed in calculations or other problem-solving operations, particularly by a computer.

However, what precisely should be envisioned under this definition, and how can the concept of an algorithm be delineated? Turing and Church provided answers to these questions in the 1930s.

In 1936, Alonzo Church introduced lambda calculus in his article [9], laying the foundation for functional programming. Lambda calculus can be understood as a universal programming language with simple rules. Church further delved into the problems that could be solved using this language and the essence of the term "problem". Following him, Alan Turing presented the theoretical model of a computer known as the Turing machine (1936) in [34]. It is not a computer as we know it today but a simple model with computational power equivalent to our real computer. It can be described as an infinite tape with a control unit. The head of the control unit reads symbols on the tape and, based on a finite set of predetermined rules, decides whether to overwrite the symbol and whether to stay put or to move to the left or to the right.

Church in [9] proved that lambda calculus has the same computational power as functions computable by an algorithm. On the other hand, Turing in [35] proved that Turing Machines have the same computational power as lambda calculus. This leads to the so-called Church-Turing Thesis: any formal model of an algorithm will be equivalent to one of the approaches of either Church or Turing. For an overview of the complex history of computability we refer to [31].

Thanks to the introduction of the Turing machine, we can distinguish problems of two types. There are problems that a Turing machine can solve, and there are problems that, no matter how hard we try, it cannot solve. Therefore, it is not fundamentally possible to find an algorithm to solve such problems. This implies that even more powerful computers in the future will not make these problems solvable, unless the formal definition of an algorithm changes significantly.

The first group is of interest to us: the problems that we can solve. This means they are computable — it might take a long time, but given a potentially infinite amount of time and memory, we can solve them. Merely being able to solve a problem may not always be sufficient, because it is also important to us to know how long it will take. To avoid incorrect results due to computer speed, memory size, etc., we will relate everything back to one standard: the Turing machine.

Stephen Cook delved deeper into these issues, classifying problems in the early 1970s into problems which we can solve quickly in principle (P problems) and problems which we can solve slowly in principle (NP problems). In this chapter, we will look closer at the division of solvable problems based on complexity and introduce the three basic complexity classes. For more details we refer to [16].

Figure 3.1: Alan Mathison Turing (1912 – 1954), picture taken from [42]

Figure 3.2: Stephen Arthur Cook (*1939), picture taken from [43]

3.1 Basic complexity classes: P, NP and NP-hard problems

Let us first informally examine groups of problems, known as the complexity classes. The "easiest" problems are P problems (polynomial), which we can solve in polynomial time. Examples include well-known algorithms like bubble sort or matrix multiplication, for more information we refer to [13]. Another group consists of NP problems (non-deterministic polynomial), such as the optimisation problem of the travelling salesman. More about this problem is in [18]. The next group comprises NP-hard problems, which are "at least as hard as the hardest problems in NP." Here, we find problems like ℓ_0 optimisation, which we will address in Section 4.4. The final group consists of problems that are both in NP and NP-hard, and we refer to them as NP-complete.

Now, we will formally define all these important terms. In order to be able to have a common appearance of problems, we focus on decision problems.

3.1.1 Definition A set X of strings in a fixed alphabet is called a *decision problem*, and $s \in X$ is called an *instance* of X.

We say that an algorithm A with outputs 'yes' and 'no' (i.e., a "program" A) *solves* X, whenever

$$A(s) = \begin{cases} \text{'yes'}, & \text{if } s \in X, \\ \text{'no'}, & \text{if } s \notin X. \end{cases}$$

Definition 3.1.1 is rather abstract. In the following example we indicate how the usual problem of multiplying two square matrices can be seen as a decision problem.

3.1.2 Example (Matrix multiplication as a decision problem) Given two matrices $\mathbf{A}$, $\mathbf{B}$, both with n columns and n rows. There is a standard linear algebra algorithm of computing the product $\mathbf{B} \cdot \mathbf{A}$. In order to see the above algorithm as a decision problem, we need a slight reformulation:

Three $n \times n$ matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}$ are given. Decide whether the equality $\mathbf{B} \cdot \mathbf{A} = \mathbf{C}$ holds.

Of course, in order that the decision-problem reformulation of matrix multiplication fully conforms with Definition 3.1.1, we would have to formally introduce the alphabet and the corresponding set X. We will not go into these details, as the relevant procedure is self evident.

3.1.3 Definition An algorithm A runs in *polynomial time* if, for every string s, $A(s)$ terminates in at most $p(\text{length}(s))$ steps, where p is some polynomial function.

3.1.4 Definition Define P to be the set of all decision problems X for which there exists an algorithm that solves X in polynomial time.

3.1.5 Example (Matrix multiplication as a problem in P) An example of a problem in P is matrix multiplication (in its decision-problem reformulation, see Example 3.1.2). For example, given matrices

$$\mathbf{A} = \begin{bmatrix} 2 & 2 \\ 3 & 5 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} -1 & 3 \\ 2 & 0 \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 7 & 12 \\ 4 & 8 \end{bmatrix}$$

we need to decide whether $\mathbf{C} = \mathbf{B} \cdot \mathbf{A}$. This is done in two steps:

(1) One computes

$$\mathbf{BA} = \begin{bmatrix} -1 & 3 \\ 2 & 0 \end{bmatrix} \cdot \begin{bmatrix} 2 & 2 \\ 3 & 5 \end{bmatrix} = \begin{bmatrix} (-1) \cdot 2 + 3 \cdot 3 & (-1) \cdot 4 + 3 \cdot 5 \\ 2 \cdot 2 + 0 \cdot 3 & 2 \cdot 4 + 0 \cdot 5 \end{bmatrix} = \begin{bmatrix} 7 & 11 \\ 4 & 8 \end{bmatrix}$$

using the standard algorithm of linear algebra.

(2) Decide whether the matrix $\mathbf{C}$ is equal to $\begin{bmatrix} 7 & 11 \\ 4 & 8 \end{bmatrix}$ obtained in step (1). One does that by comparing the individual components of $\mathbf{C}$ and $\begin{bmatrix} 7 & 11 \\ 4 & 8 \end{bmatrix}$. In our example, the answer is 'no'.

Clearly, given general matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, all with n rows and n columns, the number of steps in (1) and (2) required to finish them is bounded by a polynomial in n. More in detail:

(1) To multiply $\mathbf{A}$ and $\mathbf{B}$, we first take the 1st row of the matrix and systematically multiply each of its elements with the elements in the 1st column of the second matrix. Then we proceed to the next row of the matrix and repeat the process. This involves nested cycles. In general, if we have two matrices and one of them has dimensions $m \times n$ and the other $n \times l$, the computational complexity of their multiplication would be $m \times n \times l$. In our example, the matrices are square of size n so the computational complexity of multiplying these matrices would be n^3.

(2) To decide whether $\mathbf{B} \cdot \mathbf{A} = \mathbf{C}$ holds, we have to make n^2 comparisons to go through all components of $\mathbf{BA}$ and $\mathbf{C}$.

Thus, the decision-problem reformulation of matrix multiplication is a P-problem. Its computational complexity is $n^3 + n^2$, which is asymptotically n^3.

3.1.6 Definition An algorithm C is a *certifier* for a problem X if, for every string s, it holds: $s \in X$ if and only if there exists a string t (called certificate) such that $C(s, t) =$ 'yes'.

An example of a certifier is point (2) of the decision-problem reformulation of matrix reformulation in Example 3.1.5. In fact, it is a *polynomial* certifier, since point (2) requires a polynomial number of steps to finish.

3.1.7 Definition Define NP to be the class of all decision problems X for which there exists a certifier C such that the following two conditions hold:

(1) C is a polynomial algorithm.

(2) Every certificate t of X is of polynomial size, i.e., $\text{length}(t) \leq p(\text{length}(s))$ for some polynomial function p.

3.1.8 Definition We say that a problem X *polynomially transforms* to a problem Y (notation: $X \leq_P Y$) if given an instance x of X, we can construct an instance y of Y such that the following two conditions hold:

(1) x is a 'yes'-instance of X if and only if y is a 'yes'-instance of Y.

(2) Every certificate t of X is of polynomial size, i.e., $\text{length}(t) \leq p(\text{length}(s))$ for some polynomial function p.

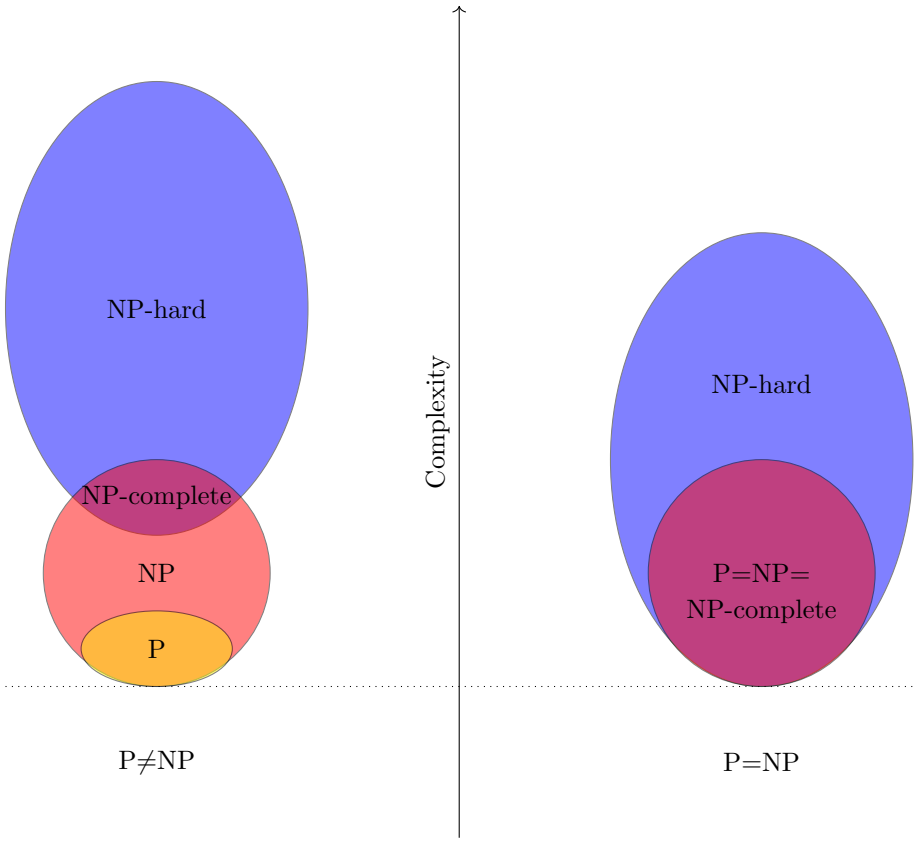
3.1.9 Definition We say that a problem Y is

(1) *NP-hard*, if for every X in NP we have $X \leq_P Y$.

(2) *NP-complete*, if Y is in NP and it is NP-hard.

3.1.10 Remark We will not give an instance of an NP-complete problem with a full proof in this text. However, in Section 4.4 below, we rely on the fact that The Exact 3-Cover Problem 4.4.2 is NP-complete. In fact, there exist quite a few NP-complete problems. For an overview of NP-complete problems we refer to [16].

3.1.11 Remark The P versus NP problem is a big unsolved problem in theoretical computer science that was posed in 1971 by Stephen Cook in his paper [12]. In computational complexity theory, it holds that P is a subset of NP. However, the question remains whether P equals NP. Generally, it is assumed that this is not true, but as of now, there is no proof for this assumption. For $P = NP$ to be true, it would be necessary to find such an NP problem that could be solved in polynomial time, which has not been achieved yet. Below, there is an Euler diagram depicting the sets of problems for two cases: $P = NP$ and $P \neq NP$.



3.2 The complexity of optimisation problems

In Section 3.1 we have formulated complexity notions for *decision problems*, see Definition 3.1.1. However, the problems we will be addressing in the following text will be mainly *optimisation problems*. A typical optimisation problem has the form, see, e.g., [4].

3.2.1 ***Typical Optimisation Problem*** *(TOP)*

Find x such that condition $C(x)$ holds and $w(x)$ is minimal.

Above, $w(x)$ is a (typically nonnegative) real number measuring "optimality" of x and C is a condition that x has to satisfy. It will often be convenient to consider a "decision-variant" of (TOP), namely, to consider the following problem:

3.2.2 *(TOP-decision)*

Given C and w. Decide, whether there exists x such that $C(x)$ holds and $w(x)$ is minimal.

It is clear that $(TOP\text{-}decision)$ is a decision problem and therefore the theory of Section 3.1 applies to it. The following two observations are now clear:

(1) Suppose we have a "fast" algorithm that solves (TOP). More in detail, suppose that we have a "fast" way of finding x that solves (TOP) or of knowing that (TOP) has no solution at all. Then we have obtained a "fast" algorithm solving $(TOP\text{-}decision)$.

(2) Suppose that we know that $(TOP\text{-}decision)$ is a "hard" problem. That is, suppose that $(TOP\text{-}decision)$ is either NP-complete or NP-hard. Then the problem (TOP) is also a "hard" problem. Were it otherwise, we could have employed the reasoning of item (1) above to conclude that $(TOP\text{-}decision)$ admits a "fast" algorithm.

The above two observations are deliberately written informally. We will see their formal instances in the following text. For example, in Theorem 4.4.6 we will claim that a certain optimisation problem (SAS) 4.4.1 is NP-hard. By that we mean that the *decision variant* of this optimisation problem is NP-hard. By the above observation this means that there is no hope for a "fast" solution of (SAS).

Chapter 4

An introduction to ℓ_p norms optimisation

In Chapter 3, we demonstrated the existence of problems classified as P and NP-hard. P problems are easy to solve, we can solve them in polynomial time, while NP-hard problems are difficult to solve. In Section 2.3, we formulated an optimisation problem that will be the focus of interest in this thesis. To delve deeper into this, it is necessary to define some terms.

The first part of this chapter explores what an ℓ_p norm is and its relationship to the Euclidean norm ℓ_2 we are familiar with. We then examine the problem of minimising the ℓ_2 norm, a relatively common problem. We show that it can be solved in polynomial time. However, as mentioned in Section 2.3, the primary focus for us is on the ℓ_0 norm. Its minimisation is discussed in the end of this chapter, and we demonstrate that this problem, unfortunately, is NP-hard and, therefore, difficult to solve.

4.1 ℓ_p norm

In this section, we delve into the so-called ℓ_p norms. To understand the concept of ℓ_p norms better, let us recall what a norm is. Recall that a vector space is just a set of vectors equipped with operations (vector addition and multiplication by a scalar) that satisfy rather straightforward axioms (we will not present the formal definition, we refer to [33]). However, to speak about a "size" of a vector, one has to endow our vector space with an *extra* structure, called a *norm of a vector*.

4.1.1 Definition Let V be a vector space over $\mathbb{R}$. The function $\|-\|\colon V \to \mathbb{R}$ satisfying the following three conditions:

(1) $\|\mathbf{v}\| \geq 0$ for any vector $\mathbf{v}$, equality holds only if $\mathbf{v} = \mathbf{o}$

(2) $\|a\mathbf{v}\| = |a| \cdot \|\mathbf{v}\|$ for any scalar a and any vector $\mathbf{v}$

(3) $\|\mathbf{v} + \mathbf{w}\| \leq \|\mathbf{v}\| + \|\mathbf{w}\|$ for any vectors $\mathbf{v}$ and $\mathbf{w}$

is called a *norm*.

An important group of norms are ℓ_p norms, see the definition below.

4.1.2 Definition Let $\mathbb{R}^n$ be the usual vector space over $\mathbb{R}$ and let $p \geq 1$ be fixed. The ℓ_p *norm* $\|\mathbf{x}\|_p$ of a vector $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$ in $\mathbb{R}^n$ is defined as

$$\|\mathbf{x}\|_p = \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p}.$$

We should prove that Definition 4.1.2 is correct, i.e., that the assignment $\mathbf{x} \mapsto (\sum_{i=1}^n |x_i|^p)^{1/p}$ satisfies the axioms of Definition 4.1.1. We do not give such a proof, we refer, e.g., to [30].

4.1.3 Examples ℓ_p norms for $p \in \{1, 2\}$:

(1) The ℓ_2 norm is a norm, also known as the *Euclidean norm*, representing the magnitude of a vector, is defined as follows

$$\|\mathbf{x}\|_2 = \sqrt{x_1^2 + x_2^2 + \ldots + x_n^2}.$$

It is induced by the standard scalar product

$$\|\mathbf{x}\|_2 = \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle} \text{ , where } \langle \mathbf{x}, \mathbf{y} \rangle = \sum_{i=1}^n x_i y_i.$$

(2) The ℓ_1 norm is a norm, also known as the *Manhattan norm*. It is computed as follows

$$\|\mathbf{x}\|_1 = |x_1| + |x_2| + \ldots + |x_n|.$$

4.1.4 Remark Another example on $\mathbb{R}^n$ norm is the so-called ℓ_∞ norm, also known as the *max-norm*. It is computed as follows

$$\|\mathbf{x}\|_\infty = \max\{|x_1|, |x_2|, \ldots, |x_n|\}.$$

In the definition of the ℓ_p norm, we have the condition $p \geq 1$. However, such ℓ_p norm will not be of much interest in this paper. But what if we tried to apply the same formula as in Definition 4.1.2 but for $0 < p < 1$? Let us choose for example $p = \frac{1}{2}$. At first sight, it may seem that the formula "worked well", but upon closer look, we find that the result does not satisfy the conditions of the norm definition, see the following example. Thus, "ℓ_p norms for $p < 1$" are *not* honest norms.

4.1.5 Example

For example, the "$\ell_{0.5}$ norm" on $\mathbb{R}^n$ defined by the formula

$$\|\mathbf{x}\|_p = \left(\sum_{i=1}^n |x_i|^p \right)^{1/p} \overset{p=\frac{1}{2}}{=} \|\mathbf{x}\|_{1/2} = \left(\sum_{i=1}^n |x_i|^{1/2} \right)^2$$

is not a norm on $\mathbb{R}^n$ in the sense of Definition 4.1.1. Indeed, condition (3) of Definition 4.1.1 is not satisfied. For example, take vectors $\mathbf{u}_1, \mathbf{v}_2 \in \mathbb{R}^2$

$$\mathbf{u} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \mathbf{v} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

Then $\|\mathbf{u} + \mathbf{v}\|_{1/2} = 4$, while $\|\mathbf{u}\|_{1/2} = \|\mathbf{v}\|_{1/2} = 1$. Hence, the inequality $\|\mathbf{u} + \mathbf{v}\|_{1/2} \leq \|\mathbf{u}\|_{1/2} + \|\mathbf{v}\|_{1/2}$ does not hold.

A similar reasoning can be used to show that the formula

$$\|\mathbf{x}\|_p = \left(\sum_{i=1}^n |x_i|^p \right)^{1/p}$$

for $0 < p < 1$ does not satisfy condition (3) of Definition 4.1.1, for any $n \geq 2$.

Although (as the previous example shows) "ℓ_p norms" are not honest norms on $\mathbb{R}^n$ for $0 < p < 1$, we will stick to the usual abuse of language and speak about ℓ_p norms even for $0 < p < 1$. Moreover, we will now introduce yet another formula that does not give a norm on $\mathbb{R}^n$ and call it the ℓ_0 norm.

4.1.6 Definition Let $\mathbb{R}^n$ be the usual vector space over $\mathbb{R}$. For a vector $\mathbf{x}$ in $\mathbb{R}^n$ we define

$$\operatorname{supp}(\mathbf{x}) = \{i \mid x_i \neq 0\}$$

and call it a *support* of $\mathbf{x}$. The ℓ_0 *norm* $\|\mathbf{x}\|_0$ of a vector $\mathbf{x}$ is defined by

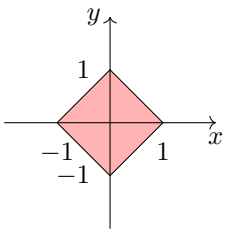
$$\|\mathbf{x}\|_0 = \operatorname{card}(\operatorname{supp}(\mathbf{x})).$$

4.1.7 Remark The assignment $\mathbf{x} \mapsto \|\mathbf{x}\|_0$ is not a norm in the sense of Definition 4.1.1. For example, in $\mathbb{R}^2$ we have $2 = \| \begin{bmatrix} 3 \\ 3 \end{bmatrix} \|_0 \neq 3 \cdot \| \begin{bmatrix} 1 \\ 1 \end{bmatrix} \|_0 = 6$. Analogous reasoning works for any $\mathbb{R}^n$, where $n \geq 1$.

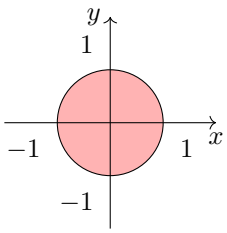
4.1.8 Definition The vector $\mathbf{x}$ in $\mathbb{R}^n$ is called *s-sparse* if at most s of its entries are non-zero, i.e., whenever

$$\|\mathbf{x}\|_0 \leq s.$$

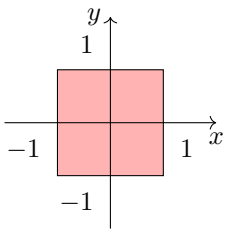
4.1.9 Examples In the context of vector norms, we are often interested in *unit balls*. In our case, we will illustrate the 2-dimensional unit balls of ℓ_p norms. Our set will be $\{\mathbf{x} \in \mathbb{R}^2 \mid \|\mathbf{x}\|_p \leq 1\}$.



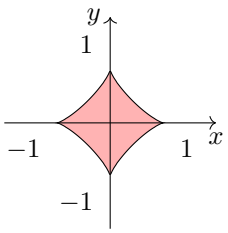
$\| \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \|_1 = |x_1| + |x_2|$



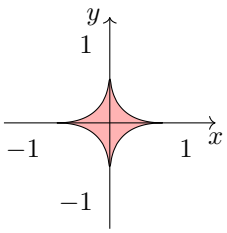
$\| \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \|_2 = \sqrt{(x_1)^2 + (x_2)^2}$



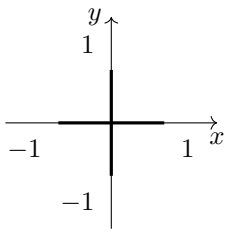
$\| \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \|_\infty = \max\{|x_1|, |x_2|\}$



$\| \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \|_{0.8} = (|x_1|^{0.8} + |x_2|^{0.8})^{1/0.8}$



$\| \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \|_{0.5} = (|x_1|^{0.5} + |x_2|^{0.5})^{1/0.5}$



$\| \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \|_0 = \text{card}(\text{supp}(\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}))$

4.2 The matrix calculus

In Section 4.3 below we will address a minimisation problem for ℓ_2 norm, We choose to demonstrate this problem for two reasons:

(1) The minimisation problem for ℓ_2 norm is similar in phrasing to our main goal: the study of minimisation problems for the ℓ_0 norm.

(2) The minimisation problem for ℓ_2 norm — as opposed to that for ℓ_0 norm — has a neat algebraic solution.

In preparation for ℓ_2 norm minimisation we need to recall various concepts from the *matrix calculus*. The topic of matrix calculus is extensive; only a small portion relevant to our needs will be covered in this thesis. Since it is not the main focus of this work, precise definitions will not be provided. For these, other sources may be useful, such as [1] or [7]. We assume that a derivative of a function of single variable is a concept that does not require an explanation in this paper, let us explore how to handle situations where vectors and matrices are involved in derivatives.

Partial derivative
Partial derivative of a function $f : \mathbb{R}^n \to \mathbb{R}$ with respect to the i-th variable at the point $\mathbf{x} = (x_1, \ldots, x_n)$ is computed by considering all variables $x_j, j \neq i$, as constants and differentiating the function with respect to one variable x_i. We denote the partial derivative of $f(\mathbf{x})$ with respect to x_i by $\frac{\partial f(\mathbf{x})}{\partial x_i}$.

For example, given the function $f(x,y) = \cos(x^2+y)$, then

$$\frac{\partial f}{\partial x} = -\sin(x^2+y)\cdot 2x$$
$$\frac{\partial f}{\partial y} = -\sin(x^2+y).$$

More generally, for vector spaces U, V of dimension n and a function $f : U \times V \to \mathbb{R}$, $(\mathbf{u},\mathbf{v}) \mapsto f(\mathbf{u},\mathbf{v})$, we define

$$\frac{\partial f}{\partial \mathbf{u}} = \begin{bmatrix} \frac{\partial f}{\partial u_1} & \frac{\partial f}{\partial u_2} & \cdots & \frac{\partial f}{\partial u_n} \end{bmatrix}$$
$$\frac{\partial f}{\partial \mathbf{v}} = \begin{bmatrix} \frac{\partial f}{\partial v_1} & \frac{\partial f}{\partial v_2} & \cdots & \frac{\partial f}{\partial v_n} \end{bmatrix}.$$

It may be noted that, e.g., $\frac{\partial f}{\partial \mathbf{u}}$ is a linear map from U to $\mathbb{R}$. Given a point (u_0, v_0) in $U \times V$, we have

$$\frac{\partial f}{\partial \mathbf{u}}(u_0,v_0) = \begin{bmatrix} \frac{\partial f}{\partial u_1}(u_0,v_0) & \cdots & \frac{\partial f}{\partial u_n}(u_0,v_0) \end{bmatrix} \cdot \begin{bmatrix} (u_0)_1 \\ \vdots \\ (u_0)_n \end{bmatrix}.$$

Total derivative
Now we will generalise the concept of derivatives for the map $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^m$. As mentioned before, we will not delve into the precise definition of a differentiable function using limits, or conditions under which is the function differentiable. We will focus on the form of the derivative (in this case, it will be a matrix) and on some examples that we will use later.

The Jacobian matrix,[1] also the *total derivative* of the map $\mathbf{f}$ at a point $\mathbf{x}$, is a matrix containing partial derivatives of all components of the map $\mathbf{f}$ with respect to all variables:

$$\frac{\mathrm{d}\mathbf{f}(\mathbf{x})}{\mathrm{d}\mathbf{x}} = \begin{bmatrix} \frac{\partial f_1(\mathbf{x})}{\partial x_1} & \frac{\partial f_1(\mathbf{x})}{\partial x_2} & \cdots & \frac{\partial f_1(\mathbf{x})}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_m(\mathbf{x})}{\partial x_1} & \frac{\partial f_m(\mathbf{x})}{\partial x_2} & \cdots & \frac{\partial f_m(\mathbf{x})}{\partial x_n} \end{bmatrix}.$$

Special cases of the Jacobian matrix are as follows:

(1) For $f : \mathbb{R} \to \mathbb{R}$, the Jacobian matrix is a 1×1 matrix. Its unique component is the scalar which is the derivative $f'(x)$ of function of one variable, as we know it.

(2) For $\mathbf{f} : \mathbb{R} \to \mathbb{R}^m$, the Jacobian matrix is the column vector $\mathbf{f}'(x) = \begin{bmatrix} f_1'(x) \\ \vdots \\ f_m'(x) \end{bmatrix}$.

(3) For $f : \mathbb{R}^n \to \mathbb{R}$, the Jacobian matrix is the row vector $f'(\mathbf{x}) = \begin{bmatrix} \frac{\partial f(\mathbf{x})}{\partial x_1} & \cdots & \frac{\partial f(\mathbf{x})}{\partial x_n} \end{bmatrix}$.

Chain Rule (Composite Function Rule)
The vector form of the chain rule is analogous to the rule we know from the derivative of a function of single variable. Let $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^m$ and $\mathbf{g} : \mathbb{R}^m \to \mathbb{R}^k$. Then the derivative of the composite function is the matrix product of the individual matrices, i.e.,

$$\frac{\mathrm{d}\mathbf{g}(\mathbf{f}(\mathbf{x}))}{\mathrm{d}\mathbf{x}} = \mathbf{g}'(\mathbf{f}(\mathbf{x}))\cdot\mathbf{f}'(\mathbf{x}).$$

Product Rule The vector form of the product rule is also analogous to the rule we know from the derivative of a function of single variable, however, it is necessary to pay attention to dimensions of matrices (in our case we will have vectors) to ensure that the product of vectors is defined.

In what follows, we will need only one instance of the product rule. Namely, for $\mathbf{g}, \mathbf{h} : \mathbb{R}^n \to \mathbb{R}^m$, the derivative of $f : \mathbf{x} \mapsto \mathbf{g}(\mathbf{x})^T\mathbf{h}(\mathbf{x})$ is given by the formula $f'(\mathbf{x}) = \mathbf{g}(\mathbf{x})^T\mathbf{h}'(\mathbf{x}) + \mathbf{h}(\mathbf{x})^T\mathbf{g}'(\mathbf{x})$.

[1] In some literature, the Jacobian matrix is called the Jacobian, but the term "Jacobian" in mathematical analysis is often used for the determinant of the Jacobian matrix, so one has to be careful with the terminology.

4.2.1 Examples We will use the following examples in our optimisation task.

(1) For $\mathbf{f}(\mathbf{x}) = \mathbf{x}$, the derivative $\mathbf{f}'(\mathbf{x})$ is the identity matrix $\mathbf{E}$.

(2) More generally, for $\mathbf{f}(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{b}$, the derivative $\mathbf{f}'(\mathbf{x})$ is $\mathbf{A}$.

(3) For the squared ℓ_2 norm $f(\mathbf{x}) = \mathbf{x}^T\mathbf{x}$, the derivative $f'(\mathbf{x})$ is $2\mathbf{x}^T$.

4.3 ℓ_2 norm optimisation

As mentioned in the introduction to this chapter, the way to solve the ℓ_2 minimisation problem will now be demonstrated. Let us recall that this problem is relatively common and we can show that it is a P problem. Hence, we can solve it in polynomial time. First, we introduce one useful definition that we will utilise through the entire thesis.

4.3.1 Definition For a matrix $\mathbf{A}$ we will always denote by n its number of columns and by m its number of rows. We say that $\mathbf{A}$ is *wide of full rank*, if $m \leq n$ and $\operatorname{rank}(\mathbf{A}) = m$ holds.

The task of the ℓ_2 minimisation problem is to minimise $\|\mathbf{x}\|_2$ subject to $\mathbf{A}\mathbf{x} = \mathbf{b}$, where $\mathbf{A}$ is wide of full rank.

We will minimise the squared norm because it will be simpler for us, and the solution set will not change. The problem task will therefore be: We are looking for $\mathbf{x}$ such that its squared ℓ_2 norm is minimal under the condition $\mathbf{A}\mathbf{x} = \mathbf{b}$. Therefore, we aim to minimise $\|\mathbf{x}\|_2^2$ under the given constraint and find the argument of this minimum (while the minimum value would provide us with the minimal norm, our interest lies in finding the corresponding $\mathbf{x}$). We can express this as follows:

4.3.2 ℓ_2 ***optimisation problem*** *(L2O)*

Given wide matrix $\mathbf{A}$ *of full rank and vector* $\mathbf{b}$. *Find* $\arg\min\{\|\mathbf{x}\|_2^2 \mid \mathbf{A}\mathbf{x} = \mathbf{b},\ \mathbf{x} \in \mathbb{R}^n\}$.

4.3.3 Theorem *The solution to (L2O) is given by* $\mathbf{x} = \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{b}$, *hence (L2O) is a P problem.*

Proof. The square of the norm can be calculated as $\|\mathbf{x}\|_2^2 = \mathbf{x}^T\mathbf{x}$, and we can rewrite the condition as $\mathbf{b} - \mathbf{A}\mathbf{x} = \mathbf{o}$. Thus, the problem (*L2O*) is a *constrained extremum* problem

$$\text{Minimise } \mathbf{x}^T\mathbf{x} \text{ subject to } \mathbf{b} - \mathbf{A}\mathbf{x} = \mathbf{o}.$$

We use the standard approach: we construct the Lagrange function

$$\mathscr{L}(\mathbf{x}, \boldsymbol{\lambda}) = \mathbf{x}^T\mathbf{x} + \boldsymbol{\lambda}^T(\mathbf{b} - \mathbf{A}\mathbf{x}),$$

where $\boldsymbol{\lambda}$ is a vector of Lagrange multipliers.

To find stationary points, we set the partial derivatives with respect to $\mathbf{x}$ and $\boldsymbol{\lambda}$ equal to $\mathbf{o}$.

$$\frac{\partial \mathscr{L}}{\partial \mathbf{x}} = 2\mathbf{x}^T - \boldsymbol{\lambda}^T\mathbf{A} \stackrel{!}{=} \mathbf{o} \tag{4.1}$$

$$\frac{\partial \mathscr{L}}{\partial \boldsymbol{\lambda}} = \mathbf{b}^T - \mathbf{x}^T\mathbf{A}^T \stackrel{!}{=} \mathbf{o} \tag{4.2}$$

From (4.1) we express $\mathbf{x}$:

$$\mathbf{x}^T = \frac{1}{2}\boldsymbol{\lambda}^T\mathbf{A}$$

or, equivalently,

$$\mathbf{x} = \frac{1}{2}\mathbf{A}^T\boldsymbol{\lambda}. \tag{4.3}$$

We modify the equation (4.2)

$$\mathbf{b}^T - \mathbf{x}^T\mathbf{A}^T = \mathbf{o}$$

or, equivalently,

$$\mathbf{b} - \mathbf{A}\mathbf{x} = \mathbf{o}$$

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

Now we substitute the expression (4.3) for $\mathbf{x}$, thus obtaining

$$\mathbf{b} = \frac{1}{2}\mathbf{A}\mathbf{A}^T\boldsymbol{\lambda}.$$

From this, we express $\boldsymbol{\lambda}$. The next step would be

$$\boldsymbol{\lambda} = 2(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{b}, \tag{4.4}$$

but we have to prove $(\mathbf{A}\mathbf{A}^T)^{-1}$ exists. We use the following two standard results.

(1) *For every matrix* $\mathbf{A}$*, the equality* $\operatorname{rank}(\mathbf{A}) = \operatorname{rank}(\mathbf{A}^T)$ *holds.*

(2) *For every matrix* $\mathbf{A}$ *with* n *columns and* m *rows, the equality* $\operatorname{rng}(\mathbf{A}^T\mathbf{A}) = \operatorname{rng}(\mathbf{A}^T)$ *holds, where* $\operatorname{rng}(\mathbf{A})$ *is the range of matrix* $\mathbf{A}$ *defined as* $\operatorname{rng}(\mathbf{A}) = \{\mathbf{A}\mathbf{x} \mid \mathbf{x} \in \mathbb{R}^n\}$.

For the proofs of the above two statements we refer to, for example, the book [33].

By combining these two statements, we obtain $\operatorname{rank}(\mathbf{A}\mathbf{A}^T) = \operatorname{rank}(\mathbf{A}) = \operatorname{rank}(\mathbf{A}^T) = \operatorname{rank}(\mathbf{A}^T\mathbf{A})$. Since we assumed at the beginning that $\mathbf{A}$ has full rank, the matrix $\mathbf{A}\mathbf{A}^T$ has m columns and m rows and has also full rank m. This implies that $\mathbf{A}\mathbf{A}^T$ has linearly independent rows, is regular, and thus, there exists an inverse $(\mathbf{A}\mathbf{A}^T)^{-1}$.

Combination of (4.3) and (4.4) will give us the stationary point

$$\mathbf{x} = \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{b},\ \boldsymbol{\lambda} = 2(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{b} \tag{4.5}$$

of the Lagrange function.

By Section 20.6 and Theorem 22.8 of [8], the above stationary point in fact yields the global minimum of $\mathbf{x} \mapsto \mathbf{x}^T\mathbf{x}$ under the constraint $\mathbf{A}\mathbf{x} = \mathbf{b}$. In other words,

$$\mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{b} = \arg\min\{\|\mathbf{x}\|_2^2 \mid \mathbf{A}\mathbf{x} = \mathbf{b}\}.$$

The expression $\mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{b}$ can be computed in polynomial time (use, e.g., standard algorithms from linear algebra). Thus, the problem (*L2O*) is a *P* problem.[2] ■

4.3.4 Remark The expression $\mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}$ from Theorem 4.3.3 is related to the concept of the *pseudoinverse* of a matrix.

More in detail, there exists a concept of a Moore-Penrose pseudoinverse $\mathbf{A}^+$ of a matrix $\mathbf{A}$, see, for example, [3] for the theory.

In case of a wide matrix $\mathbf{A}$ with full rank, we have

$$\mathbf{A}^+ = \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}.$$

Thus, the problem of minimising the ℓ_2 norm of $\mathbf{x}$ under the constraint $\mathbf{A}\mathbf{x} = \mathbf{b}$ can be expressed as $\mathbf{x} = \mathbf{A}^+\mathbf{b}$.

A similar formula, namely $\mathbf{A}^+ = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T$, holds for *tall* matrices $\mathbf{A}$. Here, $\mathbf{A}$ is a tall matrix, if it has n columns, m rows, $m \geq n$ and $\operatorname{rank}(\mathbf{A}) = n$.

For a *general* matrix $\mathbf{A}$ no neat formula for $\mathbf{A}^+$ exists; one has to use, for example, the Singular Value Decomposition of a matrix.

[2] Strictly speaking, one has to reformulate (*L2O*) as a decision problem. In fact, the decision variant of (*L2O*) is straightforward to formulate: Given $\mathbf{A}, \mathbf{b}$, decide whether there exists $\mathbf{x}$ such that $\|\mathbf{x}\|_2^2$ is minimal subject to $\mathbf{A}\mathbf{x} = \mathbf{b}$. See Section 3.2.

4.4 ℓ_0 norm optimisation

As demonstrated in Section 4.3, the minimisation of the ℓ_2 norm is a P problem. Now, we will turn our attention to the ℓ_0 norm. In 1995, it was shown that the minimisation of the ℓ_0 norm is NP-hard [23] (the concept of NP-hard problems is explained in Definition 3.1.9), and we will now examine this proof. The formulation of such a problem looks like as follows:

$$\text{Find } \mathbf{x} \text{ subject to } \|\mathbf{x}\|_0 \text{ is minimal and } \mathbf{A}\mathbf{x} = \mathbf{b}.$$

In fact, we consider a more general problem:

4.4.1 ***Sparse Approximate Solutions*** *(SAS)*

Given $0 \leq \eta < 1$, matrix $\mathbf{A}$ and vector $\mathbf{b}$. Find $\mathbf{x}$ subject to $\|\mathbf{x}\|_0$ is minimal and $\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2 \leq \eta$.

First, let us examine the Exact 3-Cover Problem. Its relevance will be demonstrated shortly.

4.4.2 ***Exact 3-Cover Problem*** *(X3C)*

Given set X with $\operatorname{card}(X) = 3q$, where $q \in \mathbb{N}$ (i.e., the number of elements in X is divisible by 3). Let $\mathcal{C}$ be a system of 3-element subsets of X. The problem is stated as follows:
Does there exist $\mathcal{C}' \subseteq \mathcal{C}$ such that $\mathcal{C}'$ is an exact cover of X? This means that we need $\mathcal{C}'$ to cover X, and, each $x \in X$ is exactly in one element of $\mathcal{C}'$.

The following statement is provided without proof; for a proof, we refer to [16].

4.4.3 Proposition *The (X3C) problem is NP-complete.*

Our strategy is as follows: we will demonstrate that the (*X3C*) problem can be polynomially reduced to the (*SAS*) problem. Thanks to Proposition 4.4.3, we know that every *NP* problem can be polynomially reduced to (*X3C*). Therefore, every *NP* problem can be reduced to (*X3C*) in polynomial time, and (*X3C*) can be polynomially reduced to (*SAS*). This implies that every *NP* problem can be polynomially reduced to (*SAS*), making (*SAS*), according to the Definition 3.1.9, *NP*-hard. Observe that we have not formulated (*SAS*) as a decision problem. Therefore, strictly speaking, we need to work with its decision-problem reformulation in Theorem 4.4.6 below.

4.4.4 *(SAS-decision)*

Given $0 \leq \eta < 1$, matrix $\mathbf{A}$ and vector $\mathbf{b}$. Decide, whether there exists $\mathbf{x}$ such that $\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2 \leq \eta$ holds and $\|\mathbf{x}\|_0$ is minimal.

Before we start proving our main result of this section, we need the following easy lemma.

4.4.5 Lemma *Suppose $\mathcal{C}' = \{C_i \mid i \in I'\}$ is an exact cover of the set X with $\operatorname{card}(X) = 3q$. Then $\mathcal{C}'$ contains precisely q elements; i.e., $\operatorname{card}(I') = q$.*

Proof. The equalities

$$3q = \operatorname{card}(X) = \sum_{i \in I'} \operatorname{card}(C_i) = \operatorname{card}(I') \cdot 3$$

hold. Thus, $\operatorname{card}(I') = \frac{3q}{3} = q$. ■

Now we formulate the following theorem we will aim to prove.

4.4.6 Theorem *The (X3C) problem can be polynomially reduced to the problem (SAS-decision). Therefore, the (SAS-decision) problem is NP-hard.*

PROOF. Let us begin by making preliminaries for the proof; we will find it useful shortly.

(1) Suppose a set X is given, with $\text{card}(X) = 3q$. Denote $3q = m$. Suppose further that

$$\mathcal{C} = \{C_i \mid i = 1, 2, \ldots, n\}$$

is a system of 3-element subsets of X.We have given the input data of the problem $(X3C)$.

(2) Let $0 \leq \eta < 1$ be given.

(3) Now, define the matrix $\mathbf{A}$ with n columns and m rows

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}_1 & \mathbf{a}_2 & \cdots & \mathbf{a}_n \end{bmatrix},$$

where vectors $\mathbf{a}_i, i = \{1, 2, \ldots, n\}$, are defined as follows:

$$(\mathbf{a}_i)_j = \begin{cases} 1, & \text{if } j \in C_i, \\ 0, & \text{otherwise.} \end{cases}$$

This means that each vector $\mathbf{a_i}$ contains exactly three ones, and the rest of the entries are zeroes.

(4) Create the column vector $\mathbf{b} = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}$ with m components.

Notice that matrix $\mathbf{A}$ and vector $\mathbf{b}$ can be constructed in polynomial time.

(5) Let $\|\mathbf{Az} - \mathbf{b}\|_2 \leq \eta$ with η given as above. Then $\|\mathbf{z}\|_0 \geq m/3$. To prove this, we need to do the following:

(i) Let $\|\mathbf{Az} - \mathbf{b}\|_2 \leq \eta$. Then $\|\mathbf{Az}\|_0 = m$.

Define the column vector $\mathbf{z} = \begin{bmatrix} z_1 \\ \vdots \\ z_n \end{bmatrix}$. For easier computation, square both sides of the inequality and expand the expression $\|\mathbf{Az} - \mathbf{b}\|_2^2$ component-wise.

$$\|\mathbf{Az} - \mathbf{b}\|_2^2 = \left(\sum_{j=1}^{m} \left(\sum_{i=1}^{n} z_i (\mathbf{a}_i)_j - 1 \right) \right)^2.$$

Suppose that $\|\mathbf{Az}\|_0 < m$. Thus, $\mathbf{Az}$ has at least one zero component. For example, if the first component of $\mathbf{Az}$ is zero, then the above equality takes the form

$$\|\mathbf{Az} - \mathbf{b}\|_2^2 = (0-1)^2 + \left(\sum_{j=2}^{m} \left(\sum_{i=1}^{n} z_i (\mathbf{a}_i)_j - 1 \right) \right)^2.$$

The value of $(0-1)^2$ is 1, and adding non-negative numbers to this means that a total value is greater or equal to 1, contradicting the inequality $\|\mathbf{Az} - \mathbf{b}\|_2 \leq \eta$. Thus, $\|\mathbf{Az}\|_0 = m$ as desired.

(ii) The inequality $\|\mathbf{Az}\|_0 \leq 3\|\mathbf{z}\|_0$ holds.

Indeed, each $\mathbf{a}_i$ contains exactly 3 ones; thus,

$$\mathbf{Az} = \sum_{j=1}^{n} (z_j \mathbf{a}_j) = \sum_{\substack{j \\ z_j \neq 0}} (z_j \mathbf{a}_j)$$

has at most 3 $\|\mathbf{z}\|_0$ non-zero components.

(iii) If $\|\mathbf{Az} - \mathbf{b}\|_2 \leq \eta$, then $\|\mathbf{z}\|_0 \geq m/3$.

This follows by combination of (5i) and (5ii).

(6) Choose $\eta = \frac{1}{2}$. By combining (3) and (4), we obtain: from an instance X, $\mathcal{C}$ of (*X3C*), we can construct the instance $\mathbf{A}$, $\mathbf{b}$, $\frac{1}{2}$ of (*SAS-decision*) in polynomial time. We write it as follows: $(X, \mathcal{C}) \rightsquigarrow (\mathbf{A}, \mathbf{b}, \frac{1}{2})$

Now we have prepared everything we need and will move on to the core of the proof. We will show that the problem (*X3C*) can be reduced to the problem (*SAS-decision*) in polynomial time. Review Definitions 3.1.8 and 3.1.6 because we will build on them.
As mentioned above in part (6) of the proof, the process $(X, \mathcal{C}) \rightsquigarrow (\mathbf{A}, \mathbf{b}, \frac{1}{2})$ runs in polynomial time. Therefore, according to the mentioned Definition 3.1.8, we have to prove the following:

$(X, \mathcal{C})$ is a 'yes'-instance of (*X3C*) if and only if $(\mathbf{A}, \mathbf{b}, \frac{1}{2})$ is a 'yes'-instance of (*SAS-decision*).

It is a sentence in the form of equivalence, so to prove it, we must prove these two implications:

(a) If $(X, \mathcal{C})$ is a 'yes'-instance of (*X3C*), then $(\mathbf{A}, \mathbf{b}, \frac{1}{2})$ is a 'yes'-instance of (*SAS-decision*).

(b) If $(\mathbf{A}, \mathbf{b}, \frac{1}{2})$ is a 'yes'-instance of (*SAS-decision*), then $(X, \mathcal{C})$ is a 'yes'-instance of (*X3C*).

Follow the steps below:

(a) Suppose $(X, \mathcal{C})$ is a 'yes'-instance of (*X3C*), it means there is $J' \subseteq J$ such that $\mathcal{C}' = \{C_i \mid i \in I'\}$ is an exact cover of X. Define a vector $\mathbf{x}_0$ with n components where $(\mathbf{x}_0)_i = 1$ if the i-th set in $\mathcal{C}$ is included in the "solution" of the (*X3C*) instance and $(\mathbf{x}_0)_i = 0$ otherwise. We can write:

$$(\mathbf{x}_0)_i = \begin{cases} 1, & \text{when } i \in I', \\ 0, & \text{when } i \notin I'. \end{cases}$$

Then the following hold:

(i) The equalities $\|\mathbf{x}_0\|_0 = \frac{m}{3} = q$ hold because $\operatorname{card}(I') = q$ (by Lemma 4.4.5).

(ii) Each element of X lies precisely in one element of $\mathcal{C}'$, so $\mathbf{A}\mathbf{x}_0 = \mathbf{b}$ and therefore the inequality $\|\mathbf{A}\mathbf{x}_0 - \mathbf{b}\|_2 \leq \frac{1}{2}$ holds.

(iii) We have $q \leq \|\mathbf{z}\|_0$ for any $\mathbf{z}$ with $\|\mathbf{A}\mathbf{z} - \mathbf{b}\|_2 \leq \frac{1}{2}$, we know that from part (5) of this proof. Therefore $\mathbf{x}_0$ minimises the ℓ_0 norm.
To conclude: we showed that $(\mathbf{A}, \mathbf{b}, \frac{1}{2})$ is a 'yes'-instance of (*SAS-decision*).

(b) Suppose that $(\mathbf{A}, \mathbf{b}, \frac{1}{2})$ is a 'yes'-instance of (*SAS-decision*). Denote the ℓ_0-minimal vector by $\mathbf{x}_0$. Using part (5) of this proof and since $\mathbf{A}$ has only three non-zero entries in each of its columns, the equality $\|\mathbf{x}_0\|_0 = q$ must hold. Denote by $I' \subseteq I$ the indices i, such that the i-th component x_i of $\mathbf{x}_0$ is non-zero. Thus $\mathcal{C}' = \{C_i \mid i \in I'\}$ is an exact cover of X. To conclude: $(X, \mathcal{C})$ is a 'yes'-instance of (*X3C*).

By this, our proof comes to an end; we have shown that the (*SAS-decision*) problem 4.4.4 is *NP*-hard.

■

Thus, we have shown that the (*SAS*) problem 4.4.1 is inherently a "hard" problem. See Section 3.2 for the reasoning. Since we do not want to give up on (*SAS*), we will seek a suitable replacement for (*SAS*) that is computationally more tractable. As it turns out a suitable replacement exists. We devote the next chapter to this problem.

Chapter 5

Basis Pursuit

So far, we have discussed ℓ_p norm optimisation only in general. Now, we consider it more in the context of Compressed Sensing. Therefore, we will use the notation $\mathbf{\Theta}$ for our ambient matrix (that we denoted by $\mathbf{A}$ in Chapter 4), which is a typical notation often used in the Compressed Sensing theory.

In Section 2.3, we formulated the problem we are facing as follows: we have a wide matrix $\mathbf{\Theta}$ and a vector $\mathbf{y}$ containing the measured data. We are looking for the *sparsest* vector $\mathbf{s}$ solving the equation $\mathbf{\Theta s} = \mathbf{y}$. The precise formulation of that problem is as follows:

$$\text{Given wide matrix } \mathbf{\Theta} \text{ of a full rank and vector } \mathbf{y}. \text{ Find } \arg\min\{\|\mathbf{s}\|_0 \mid \mathbf{\Theta s} = \mathbf{y},\ \mathbf{s} \in \mathbb{R}^n\}.$$

In Chapter 4, we showed that the above problem is *NP*-hard, since it is just an instance of the (*SAS*) problem 4.4.1 with $\eta = 0$. There are several strategies to deal with it, and this thesis is focused on one specific strategy: we will transform the ℓ_0 minimisation problem to an ℓ_1 minimisation problem. The conditions under which we can do this will be shown in the next chapters; now, we will demonstrate why ℓ_1 minimisation is suitable for our problem.

Note In some literature, this conversion into an ℓ_1 minimisation is called a convex relaxation. However, this term is often used only for relaxation of the constraining conditions, such as replacing $x \in \{0, 1\}$ by $x \in [0, 1]$, for example.

5.1 ℓ_1 norm optimisation

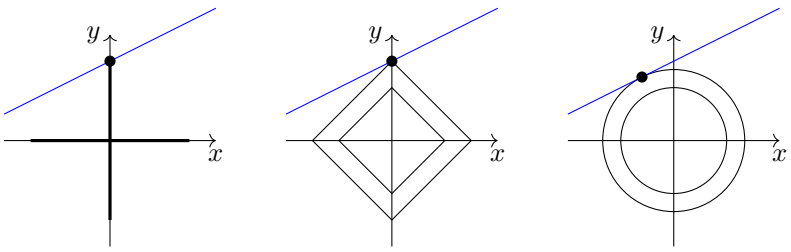
The ℓ_1 minimisation problem (often called *Basis Pursuit*) is formulated as follows:

5.1.1 ℓ_1 ***optimisation problem*** *(L1O)*

Given wide matrix $\mathbf{\Theta}$ *of a full rank and vector* $\mathbf{y}$. *Find* $\arg\min\{\|\mathbf{s}\|_1 \mid \mathbf{\Theta s} = \mathbf{y},\ \mathbf{s} \in \mathbb{R}^n\}$.

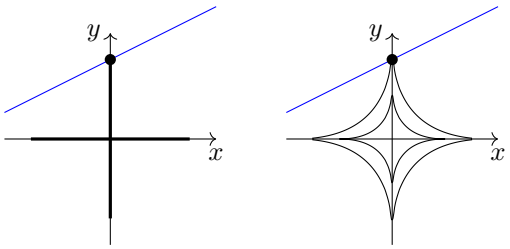
The reason why ℓ_1 minimisation is convenient is that it leads to a sparse solution. Let us look at it geometrically:

Below there are illustrated the 2-dimensional unit spheres of (from the left) ℓ_0, ℓ_1 and ℓ_2 norms, which we introduced in Chapter 4, and a line $y = ax + b$. The optimal solution minimising the ℓ_p norm and satisfying the given condition is found by "inflating" the respective unit spheres.



For both the ℓ_0 and ℓ_1 norms, the solution lies on the coordinate axis, thus we obtain a sparse solution (one component of the vector will be zero). However, for the ℓ_2 norm, we see that the optimal solution does not typically lie on any of the coordinate axes; none of the vector components will be zero, and thus the solution is not sparse. We have shown in Theorem 4.3.3 that the ℓ_2 minimisation is easy (in fact, it runs in polynomial time). However, as the pictures above indicate, the ℓ_2 minimisation is not suitable for us. It seems that ℓ_1 minimisation is a good candidate as the replacement of ℓ_0 minimisation.

Of course, we could also use ℓ_p norm minimisation for $0 < p < 1$. As one can see in the figure below, we get a sparse solution too. Unfortunately, the corresponding ℓ_p norms are not honest norms either and their minimisation is not a convex problem.



To conclude, ℓ_1 minimisation is a suitable replacement for the *NP*-hard ℓ_0 minimisation problem because it is a convex problem and it provides sparse solutions. So far, we have only indicated this geometrically. Now we will prove it.

5.1.2 Theorem *Suppose* $\boldsymbol{\Theta} = \begin{bmatrix}\boldsymbol{\theta_1} & \cdots & \boldsymbol{\theta_n}\end{bmatrix}$ *is a wide matrix of a full rank. Assume that the problem (L1O) has a unique solution* $\mathbf{s}^*$*, i.e., assume that*

$$\min\{\|\mathbf{s}\|_1 \mid \mathbf{y} = \boldsymbol{\Theta}\mathbf{s}\}$$

is attained at the unique vector s^**. Then the set* $\{\boldsymbol{\theta_j} \mid j \in \operatorname{supp}(\mathbf{s}^*)\}$ *is linearly independent and* $\|\mathbf{s}^*\|_0 \leq m$ *holds.*

Proof. We prove this theorem by contradiction. Let us denote $\operatorname{supp}(\mathbf{s}^*)$ by the letter S. Assume that the set $\{\boldsymbol{\theta_j} \mid j \in S\}$, i.e., the set of columns of $\boldsymbol{\Theta}$ corresponding to the non-zero entries of $\mathbf{s}^*$, is linearly dependent. Thus, there exists a non-zero vector $\mathbf{v}$ with $\operatorname{supp}(\mathbf{v}) = \operatorname{supp}(\mathbf{s}^*)$, such that $\boldsymbol{\Theta}\mathbf{v} = \mathbf{o}$.

Now consider the vector $\mathbf{s}^* + \varepsilon\mathbf{v}$, where $\varepsilon \neq 0$. First, we verify that it is a solution to our problem: the equalities

$$\boldsymbol{\Theta}(\mathbf{s}^* + \varepsilon\mathbf{v}) = \boldsymbol{\Theta}\mathbf{s}^* + \varepsilon\boldsymbol{\Theta}\mathbf{v} = \mathbf{y}$$

hold. Since $\mathbf{s}^*$ is the assumed unique optimal solution, i.e., the solution with the smallest ℓ_1 norm, the inequality

$$\|\mathbf{s}^*\|_1 < \|\mathbf{s}^* + \varepsilon\mathbf{v}\|_1.$$

holds. Expanding the expression on the right-hand side from the definition of the ℓ_1 norm, we obtain

$$\|\mathbf{s}^* + \varepsilon\mathbf{v}\|_1 = \sum_{j\in S} |s_j^* + \varepsilon v_j| = \sum_{j\in S} (s_j^* + \varepsilon v_j) \cdot \operatorname{sign}(s_j^* + \varepsilon v_j).$$

If $|\varepsilon|$ is sufficiently small, the expression εv_j will not affect the sign of $s_j^* + \varepsilon v_j$. Specifically, it must hold

$$|\varepsilon| < \frac{|s_j^*|}{|v_j|} \text{ for all } j, \text{ hence } |\varepsilon| < \min \frac{|s_j^*|}{|v_j|}.$$

For such a non-zero ε, we can write

$$\|\mathbf{s}^*\|_1 < \sum_{j\in S} (s_j^* + \varepsilon v_j) \cdot \operatorname{sign}(s_j^*) = \sum_{j\in S} s_j^* \cdot \operatorname{sign}(s_j^*) + \sum_{j\in S} \varepsilon v_j \cdot \operatorname{sign}(s_j^*).$$

We can further modify the expression to get

$$\|\mathbf{s}^*\|_1 < \|\mathbf{s}^*\|_1 + \varepsilon \sum_{j\in S} v_j \cdot \operatorname{sign}(s_j^*).$$

If $\operatorname{sign}(s_j^*)$ is negative, we can choose a positive ε, and the entire expression

$$\varepsilon \sum_{j \in S} v_j \cdot \operatorname{sign}(s_j^*)$$

will be negative. Similarly, we can choose a negative ε in the opposite case. Thus, the right-hand side of the inequality will be less than the left-hand side, which contradicts the original statement. Therefore, we have proven that the set $\{\boldsymbol{\theta_j} \mid j \in \operatorname{supp}(s^*)\}$ is linearly independent.

Recall that the matrix $\boldsymbol{\Theta}$ has n columns and m rows. Because $\{\boldsymbol{\theta_j} \mid j \in \operatorname{supp}(s^*)\}$ is linearly independent and the vector $\mathbf{y}$ has m elements, the number of non-zero entries in $\mathbf{s}^*$ cannot be bigger than m. This means $\|\mathbf{s}^*\|_0 \leq m$. ■

5.2 Basis Pursuit as a linear program

The (*L1O*) problem 5.1.1 can be written as a linear program. This is very important because linear programming is a discipline of mathematical programming and there are several well-known and reliable algorithms for solving linear programs, for example the simplex method.[1] For more information on solving linear programs, we refer to [17].

In general, the linear program is formulated as the minimisation or maximisation of a linear function subject to linear constraints. Here we will use a special form, called the canonical form. It is often utilised in matrix representation:

5.2.1 ***A Linear Program*** *(LP)*

Given matrix $\mathbf{A}$ *with* n *columns and* m *rows, vectors* $\mathbf{c}$ *in* $\mathbb{R}^n$, $\mathbf{b}$ *in* $\mathbb{R}^m$. *Find*
$\arg\min\{\mathbf{c}^T\mathbf{x} \mid \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \in \mathbb{R}^n, \mathbf{x} \geq \mathbf{o}$, *(i.e., every component of* $\boldsymbol{x}$ *is non-negative)*$\}$.

The above is a minimisation of a linear function subject to linear equality constraints and non-negative variables.

We now show how to convert (*L1O*) 5.1.1 into (*LP*) 5.2.1. First, we convert the minimised function $\|\mathbf{s}\|_1$ into the form of the linear function $\mathbf{c}^T\mathbf{x}$. Let us introduce the so-called slack variables: we divide the unbounded variable $\mathbf{s}$ into two non-negative variables $\mathbf{s}^+$ and $\mathbf{s}^-$ such that $\mathbf{s} = \mathbf{s}^+ - \mathbf{s}^-$. We put all positive entries of $\mathbf{s}$ into the vector $\mathbf{s}^+$ and all negative entries into the vector $\mathbf{s}^-$ but with the positive signs, and fill the rest with zeroes. Both vectors will thus contain only non-negative entries. Let us provide an example:

$$\mathbf{s} = \begin{bmatrix} 4 \\ 2 \\ 0 \\ -3 \\ -2 \\ 0 \\ 2 \end{bmatrix} = \begin{bmatrix} 4 \\ 2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 2 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ 0 \\ 3 \\ 2 \\ 0 \\ 2 \end{bmatrix}.$$

Thus, we get

$$\|\mathbf{s}\|_1 = \sum_{i=1}^{n} |s_i| = \sum_{i=1}^{n} s_i^+ + \sum_{i=1}^{n} s_i^- = \begin{bmatrix} \mathbf{1}^T & \mathbf{1}^T \end{bmatrix} \begin{bmatrix} \mathbf{s}^+ \\ \mathbf{s}^- \end{bmatrix},$$

where $\mathbf{1}$ is the matrix with one column containing $1'$s. Now we modify the linear constraints:

$$\mathbf{y} = \boldsymbol{\Theta}\mathbf{s} = \boldsymbol{\Theta}(\mathbf{s}^+ - \mathbf{s}^-) = \begin{bmatrix} \boldsymbol{\Theta} & -\boldsymbol{\Theta} \end{bmatrix} \begin{bmatrix} \mathbf{s}^+ \\ \mathbf{s}^- \end{bmatrix}.$$

The final formulation of ℓ_1 minimisation in the form of a linear program looks like as follows:

[1] Although the simplex method — in its worst-case scenario — has exponential complexity, it was proved in [32] that it "usually" runs in polynomial time.

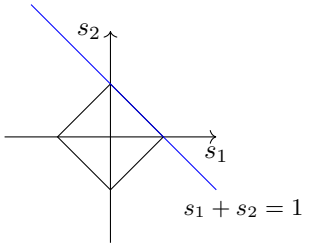
5.2.2 ℓ_1 ***optimisation as a linear program*** *(L1LP)*

Given matrix $\mathbf{\Theta}$ *with* n *columns and* m *rows and vector* $\mathbf{y}$ *in* $\mathbb{R}^m$*. Find*

$$\arg\min\left\{\begin{bmatrix}\mathbf{1}^T & \mathbf{1}^T\end{bmatrix}\begin{bmatrix}\mathbf{s}^+\\ \mathbf{s}^-\end{bmatrix} \;\middle|\; \begin{bmatrix}\mathbf{\Theta} & -\mathbf{\Theta}\end{bmatrix}\begin{bmatrix}\mathbf{s}^+\\ \mathbf{s}^-\end{bmatrix} = \mathbf{y}, \begin{bmatrix}\mathbf{s}^+\\ \mathbf{s}^-\end{bmatrix} \geq \mathbf{o}\right\}.$$

Hence we have made first steps in tackling the Compressed Sensing problem: we replaced the ℓ_0 minimisation with ℓ_1 minimisation, and, ultimately, with a problem of a linear programming. Our replacement hinges upon Theorem 5.1.2. That is, we need to assume that the solution of *(L1O)* 5.1.1 is unique, which is not the case in general.

5.2.3 Example If we want to minimise $\| \begin{bmatrix}s_1\\ s_2\end{bmatrix} \|_1$ subject to $\begin{bmatrix}1 & 1\end{bmatrix} \cdot \begin{bmatrix}s_1\\ s_2\end{bmatrix} = \begin{bmatrix}1\end{bmatrix}$, it is clear from the following picture



that this problem has at least three solutions, namely $\begin{bmatrix}1\\ 0\end{bmatrix}, \begin{bmatrix}0\\ 1\end{bmatrix}, \begin{bmatrix}1/2\\ 1/2\end{bmatrix}$.

In the following chapters we will concentrate on conditions that ensure Theorem 5.1.2 is applicable.

Chapter 6

The Null Space Property

In Chapter 5, we discussed that the *NP*-hard problem of ℓ_0 optimisation can be, under certain conditions on the ambient matrix, transformed into the ℓ_1 optimisation problem. In this chapter, we will introduce one of these conditions called the *Null Space Property* (NSP).[1] As the name says, NSP is a property of a matrix that speaks about the structure of its kernel. In fact, we show in Theorem 6.2.6 below that NSP is equivalent to the existence of a unique "sufficiently sparse" solution of (*L1O*) 5.1.1. Hence, for NSP matrices we can use the theory given in Chapter 5: the ℓ_0 minimisation can be replaced with ℓ_1 minimisation.

6.1 Nullifying components of a vector

Before we delve into the Null Space Property of a matrix, we introduce a useful technique of *nullifying* chosen components of a vector. For example, given a vector $\begin{bmatrix}-3\\2\\1\\4\end{bmatrix}$, we want to produce $\begin{bmatrix}0\\2\\0\\0\end{bmatrix}$ that has components with indices in the set $\{1,3,4\}$ nullified. Clearly, such a construction can be described by applying a certain linear map that we now define in full generality.

6.1.1 Definition Suppose $S \subseteq \{1,2,\dots,n\}$. We denote by $\mathbf{N}_S$ the $n \times n$ square matrix such that its i-th row $\mathbf{r}_i$ has the following form

$$\mathbf{r}_i = \begin{cases} \mathbf{e}_i^T, & i \in S, \\ \mathbf{o}, & i \in S^{\mathsf{c}} = \{1,2,\dots,n\} \setminus S \end{cases}$$

where, above, $\mathbf{e}_i$ denotes the i-th vector of the canonical basis of $\mathbb{R}^n$. We call $\mathbf{N}_S$ the *S-nullifying matrix*.

Thus, for example, for $S = \{2\} \subseteq \{1,2,3,4\}$ we have $S^{\mathsf{c}} = \{1,2,3,4\} \setminus \{2\} = \{1,3,4\}$ and

$$\mathbf{N}_S = \begin{bmatrix}0&0&0&0\\0&1&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}.$$

The equality

$$\begin{bmatrix}0&0&0&0\\0&1&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix} \cdot \begin{bmatrix}-3\\2\\1\\4\end{bmatrix} = \begin{bmatrix}0\\2\\0\\0\end{bmatrix}$$

holds, as expected.

It will be convenient to refer to the following trivial result.

[1] The Null Space Property was introduced in [10], the textbook account is given in [15].

6.1.2 Lemma *Let $S \subseteq \{1, 2, \dots, n\}$ and put $S^{\mathsf{c}} = \{1, 2, \dots, n\} \setminus S$. Then the following equalities*

$$\mathbf{N}_S + \mathbf{N}_{S^{\mathsf{c}}} = \mathbf{E}_n \qquad \|\mathbf{v}\|_1 = \|\mathbf{N}_S\mathbf{v}\|_1 + \|\mathbf{N}_{S^{\mathsf{c}}}\mathbf{v}\|_1$$

hold. Above, $\mathbf{E}_n$ is the $n \times n$ identity matrix and $\mathbf{v}$ is any vector in $\mathbb{R}^n$.

Proof. This first equality follows immediately from the definitions of $\mathbf{N}_S$ and $\mathbf{N}_{S^{\mathsf{c}}}$. For the second equality, consider that equalities

$$\|\mathbf{v}\|_1 = \sum_{i=1}^{n} |v_i| = \sum_{i \in S} |v_i| + \sum_{j \in S^{\mathsf{c}}} |v_j| = \|\mathbf{N}_S\mathbf{v}\|_1 + \|\mathbf{N}_{S^{\mathsf{c}}}\mathbf{v}\|_1$$

hold for any vector $\mathbf{v}$ in $\mathbb{R}^n$. ■

The impact of Lemma 6.1.2 can be seen in the following example.

6.1.3 Example Consider a vector $\mathbf{v} = \begin{bmatrix} 1 \\ 2 \\ -3 \\ 0 \\ 3 \\ 4 \\ 7 \end{bmatrix}$ and a set $S = \{1, 3, 5, 6\}$. In this case, vectors $\mathbf{N}_S \cdot \mathbf{v}$ and $\mathbf{N}_{S^{\mathsf{c}}} \cdot \mathbf{v}$ will look like

$$\mathbf{N}_S \cdot \mathbf{v} = \begin{bmatrix} 1 \\ 0 \\ -3 \\ 0 \\ 3 \\ 4 \\ 0 \end{bmatrix}, \quad \mathbf{N}_{S^{\mathsf{c}}} \cdot \mathbf{v} = \begin{bmatrix} 0 \\ 2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 7 \end{bmatrix}$$

Clearly, the equalities $\mathbf{v} = \mathbf{E}_7 \cdot \mathbf{v} = (\mathbf{N}_S + \mathbf{N}_{S^{\mathsf{c}}}) \cdot \mathbf{v} = \mathbf{N}_S\mathbf{v} + \mathbf{N}_{S^{\mathsf{c}}}\mathbf{v}$ hold. Analogously, the equalities $\|\mathbf{v}\|_1 = 20 = 11 + 9 = \|\mathbf{N}_S\mathbf{v}\|_1 + \|\mathbf{N}_{S^{\mathsf{c}}}\mathbf{v}\|_1$ hold.

6.2 The Null Space Property and sparse recovery

Now, let us give the formal definition of the Null Space Property.

6.2.1 Definition Suppose $\boldsymbol{\Theta}$ is a wide matrix of a full rank.

(1) We say that $\boldsymbol{\Theta}$ has the *Null Space Property relative to a set* $S \subseteq \{1, 2, \dots, n\}$ if

$$\|\mathbf{N}_S\mathbf{v}\|_1 < \|\mathbf{N}_{S^{\mathsf{c}}}\mathbf{v}\|_1 \quad \text{for all } \mathbf{v} \in \ker(\boldsymbol{\Theta}) \setminus \{\mathbf{o}\}.$$

(2) We say that $\boldsymbol{\Theta}$ has the *Null Space Property of order* s if it has NSP relative to any set $S \subseteq \{1, 2, \dots, n\}$ with $\operatorname{card}(S) \leq s$.

6.2.2 Example We provide an example of a matrix having NSP and of a matrix not having NSP.

(1) Consider the matrix $\mathbf{A} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix}$. Its kernel is the span of the vector $\begin{bmatrix} -1 \\ -1 \\ 1 \\ 1 \end{bmatrix}$. Then every vector $\mathbf{v} \in \ker(\mathbf{A}) \setminus \{\mathbf{o}\}$ has the form $\mathbf{v} = \begin{bmatrix} -a \\ -a \\ a \\ a \end{bmatrix}$ for $a \neq 0$. Since, for every one-element set $S \subseteq \{1, 2, 3, 4\}$ the equalities $\|\mathbf{N}_S\mathbf{v}\|_1 = |a|$ and $\|\mathbf{N}_{S^{\mathsf{c}}}\mathbf{v}\|_1 = 3|a|$ hold for all $a \neq 0$, the matrix $\mathbf{A}$ has NSP of order 1.

(2) Consider the matrix $\mathbf{A} = \begin{bmatrix} 2 & 1 & 0 \\ 2 & 0 & 1 \end{bmatrix}$. Its kernel is the span of the vector $\begin{bmatrix} -1 \\ 2 \\ 2 \end{bmatrix}$ and we denote this vector by $\mathbf{v}$. We are interested in whether the matrix $\mathbf{A}$ has NSP of order $s = 2$. Look at the sets $S_1 = \{1, 2\}$, $S_2 = \{2, 3\}$, $S_3 = \{1, 3\}$, $S_4 = \{1\}$, $S_5 = \{2\}$, and $S_6 = \{3\}$. For all these sets, the inequality $\|\mathbf{N}_S\mathbf{v}\|_1 < \|\mathbf{N}_{S^c}\mathbf{v}\|_1$ must hold.

For S_1 we have

$$\mathbf{N}_{S_1}\mathbf{v} = \begin{bmatrix} -1 \\ 2 \\ 0 \end{bmatrix}, \quad \|\mathbf{N}_{S_1}\mathbf{v}\|_1 = 1 + 2 = 3,$$

$$\mathbf{N}_{S_1^c}\mathbf{v} = \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix}, \quad \|\mathbf{N}_{S_1^c}\mathbf{v}\|_1 = 2.$$

The inequality $\|\mathbf{N}_{S_1}\mathbf{v}\|_1 < \|\mathbf{N}_{S_1^c}\mathbf{v}\|_1$ does not hold, which indicates that the matrix $\mathbf{A}$ does not have NSP of order 2.

6.2.3 Remark Notice that, in Definition 6.2.1 (2) above we can restrict ourselves to sets S with $\mathrm{card}(S) = s$. More in detail, for $\boldsymbol{\Theta}$ the following are equivalent:

(a) $\boldsymbol{\Theta}$ has NSP of order s in the sense of Definition 6.2.1 (2).

(b) $\boldsymbol{\Theta}$ has NSP relative to any set $S \subseteq \{1, 2, \ldots, n\}$ with $\mathrm{card}(S) = s$.

(c) Given $\mathbf{v} \in \ker(\boldsymbol{\Theta} \setminus \{\mathbf{o}\})$, arrange its components such that $|v_{i_1}| \geq |v_{i_2}| \geq \ldots \geq |v_{i_n}|$ holds. Put $S^* = \{i_1, i_2, \ldots, i_s\}$. Then $\|\mathbf{N}_{S^*}\mathbf{v}\|_1 < \|\mathbf{N}_{S^{*c}}\mathbf{v}\|_1$ holds.

Clearly, (a) implies (b) and (b) implies (c). To prove that (c) implies (a), fix a vector $\mathbf{v} \in \ker(\boldsymbol{\Theta}) \setminus \{\mathbf{o}\}$ and a set $S \subseteq \{1, 2, \ldots, n\}$ with $\mathrm{card}(S) \leq s$. We want to prove that $\|\mathbf{N}_S\mathbf{v}\|_1 < \|\mathbf{N}_{S^c}\mathbf{v}\|_1$ holds.

Using the set S^* of s indices where $|v_i|$ are the largest, we obtain the inequality

$$\|\mathbf{N}_S\mathbf{v}\|_1 = \sum_{i \in S} |v_i| \leq \sum_{i \in S^*} |v_i| = \|\mathbf{N}_{S^*}\mathbf{v}\|_1$$

and, therefore, also the inequality

$$\|\mathbf{N}_{S^{*c}}\mathbf{v}\|_1 = \sum_{i \in S^{*c}} |v_i| = \|\mathbf{v}\|_1 - \sum_{i \in S^*} |v_i| \leq \|\mathbf{v}\|_1 - \sum_{i \in S} |v_i| = \sum_{i \in S^c} |v_i| = \|\mathbf{N}_{S^c}\mathbf{v}\|_1.$$

Since, by assumption, the inequality

$$\|\mathbf{N}_{S^*}\mathbf{v}\|_1 < \|\mathbf{N}_{S^{*c}}\mathbf{v}\|_1$$

holds, by putting all of the above inequalities together, we obtain

$$\|\mathbf{N}_S\mathbf{v}\|_1 \leq \|\mathbf{N}_{S^*}\mathbf{v}\|_1 < \|\mathbf{N}_{S^{*c}}\mathbf{v}\|_1 \leq \|\mathbf{N}_{S^c}\mathbf{v}\|_1$$

establishing the desired inequality.

Now, we will look at an equivalent formulation of the NSP definition.

6.2.4 Proposition *Suppose $\boldsymbol{\Theta}$ is a wide matrix of a full rank, let $S \subseteq \{1, 2, \ldots, n\}$. The following are equivalent:*

(1) *$\boldsymbol{\Theta}$ has the NSP relative to S.*

(2) *The inequality $2\|\mathbf{N}_S\mathbf{v}\|_1 < \|\mathbf{v}\|_1$ holds for all $\mathbf{v} \in \ker(\boldsymbol{\Theta}) \setminus \{\mathbf{o}\}$.*

Proof. Look at the Definition 6.2.1 (1) of NSP relative to S. We will add $\|\mathbf{N}_S\mathbf{v}\|_1$ to both sides of the inequality and using the Lemma 6.1.2, we obtain the formulation (2) in Proposition 6.2.4. ■

6.2.5 Remark Suppose $\mathbf{v} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} \in \mathbb{R}^n$ and suppose the inequality $2\|\mathbf{N}_S\mathbf{v}\|_1 < \|\mathbf{v}\|_1$ holds for every set $S \subseteq \{1, 2, \dots, n\}$ such that $\text{card}(S) = s = 1$. Then it must hold that

$$2|v_k| < \sum_{i=1}^{n} |v_i| \quad \text{for all } k.$$

Hence we have

$$2\sum_{k=1}^{n} |v_k| < n \cdot \sum_{i=1}^{n} |v_i|$$

or, equivalently,

$$2\|\mathbf{v}\|_1 < n \cdot \|\mathbf{v}\|_1$$

Therefore $n > 2$ holds, i.e., a vector $\mathbf{v}$ satisfying the above must be at least from $\mathbb{R}^3$. Let us show an example of such a vector.

Given a vector $\mathbf{v} = \begin{bmatrix} 1 \\ 1.5 \\ 1 \end{bmatrix}$. See that inequality $2\|\mathbf{N}_S\mathbf{v}\|_1 < \|\mathbf{v}\|_1$ holds for any set S with $\text{card}(S) = s = 1$, because

$$\begin{aligned} 2 \cdot 1 &< 1 + 1.5 + 1 = 3.5 \\ 2 \cdot 1.5 &< 3.5 \\ 2 \cdot 1 &< 3.5 \end{aligned}$$

Now we come to an important theorem that gives us the connection between NSP and the exact recovery of sparse vectors via (*L1O*) 5.1.1.

6.2.6 Theorem *Let $\boldsymbol{\Theta}$ be a wide matrix of a full rank. Let $S \subseteq \{1, 2, \dots, n\}$. Then the following are equivalent*

(1) *If $\mathbf{x}$ is a vector with $\text{supp}(\mathbf{x}) = S$, then $\mathbf{x}$ is the unique solution of the problem $\arg\min\{\|\mathbf{s}\|_1 \mid \boldsymbol{\Theta}\mathbf{s} = \boldsymbol{\Theta}\mathbf{x}\}$.*

(2) *The matrix $\boldsymbol{\Theta}$ has NSP relative to S.*

Proof. To prove that (1) implies (2), assume that every vector $\mathbf{x} \in \mathbb{R}^n$ with $\text{supp}(\mathbf{x}) = S$ is the unique solution of $\arg\min\{\|\mathbf{s}\|_1 \mid \boldsymbol{\Theta}\mathbf{s} = \boldsymbol{\Theta}\mathbf{x}\}$.

Let $\mathbf{v}$ be a vector $\in \ker(\boldsymbol{\Theta}) \setminus \{\mathbf{o}\}$. Then, $\mathbf{N}_S\mathbf{v}$ is the unique solution of $\arg\min\{\|\mathbf{s}\|_1 \mid \boldsymbol{\Theta}\mathbf{s} = \boldsymbol{\Theta} \cdot \mathbf{N}_S\mathbf{v}\}$. We have $\mathbf{v} = \mathbf{N}_{S^c}\mathbf{v} + \mathbf{N}_S\mathbf{v}$ by Lemma 6.1.2. So it holds

$$\boldsymbol{\Theta}(-\mathbf{N}_{S^c}\mathbf{v}) = \boldsymbol{\Theta}(\mathbf{N}_S\mathbf{v} - \mathbf{v}) = \boldsymbol{\Theta}\mathbf{N}_S\mathbf{v} - \boldsymbol{\Theta}\mathbf{v}.$$

Since $\mathbf{v} \in \ker(\boldsymbol{\Theta})$, the equality $\boldsymbol{\Theta}\mathbf{v} = \mathbf{o}$ holds. We have $\boldsymbol{\Theta}(-\mathbf{N}_{S^c}\mathbf{v}) = \boldsymbol{\Theta}\mathbf{N}_S\mathbf{v}$, using Lemma 6.1.2. Moreover, $\mathbf{N}_S\mathbf{v} \neq -\mathbf{N}_{S^c}\mathbf{v}$ holds because $\mathbf{v} \neq \mathbf{o}$. Since we assumed $\mathbf{N}_S\mathbf{v}$ was the *unique* solution, we obtain the inequality $\|\mathbf{N}_S\mathbf{v}\|_1 < \| - \mathbf{N}_{S^c}\mathbf{v}\|_1 = \|\mathbf{N}_{S^c}\mathbf{v}\|_1$ that was desired.

To prove that (2) implies (1), assume that $\boldsymbol{\Theta}$ has NSP relative to S and choose a vector $\mathbf{x} \in \mathbb{R}^n$ with $\text{supp}(\mathbf{x}) = S$. We want to show that $\mathbf{x}$ is the unique solution of $\arg\min\{\|\mathbf{s}\|_1 \mid \boldsymbol{\Theta}\mathbf{s} = \boldsymbol{\Theta}\mathbf{x}\}$. Thus, choose any vector $\mathbf{z}$ with $\boldsymbol{\Theta}\mathbf{x} = \boldsymbol{\Theta}\mathbf{z}$ and $\mathbf{z} \neq \mathbf{x}$. We want to prove that $\|\mathbf{x}\|_1 < \|\mathbf{z}\|_1$ holds.

Let $\mathbf{v} = \mathbf{x} - \mathbf{z}$. Because $\boldsymbol{\Theta}\mathbf{v} = \boldsymbol{\Theta}(\mathbf{x} - \mathbf{z}) = \boldsymbol{\Theta}\mathbf{x} - \boldsymbol{\Theta}\mathbf{z}$ and $\boldsymbol{\Theta}\mathbf{x} = \boldsymbol{\Theta}\mathbf{z}$, we obtain $\boldsymbol{\Theta}\mathbf{v} = \mathbf{o}$, implying $\mathbf{v} \in \ker(\boldsymbol{\Theta}) \setminus \{\mathbf{o}\}$. Using the triangle inequality, we have

$$\|\mathbf{x}\|_1 = \|\mathbf{x} - \mathbf{N}_S\mathbf{z} + \mathbf{N}_S\mathbf{z}\|_1 \leq \|\mathbf{x} - \mathbf{N}_S\mathbf{z}\|_1 + \|\mathbf{N}_S\mathbf{z}\|_1.$$

Due to the NSP, the inequality $\|\mathbf{x} - \mathbf{N}_S\mathbf{z}\|_1 < \|\mathbf{N}_{S^c}(\mathbf{x} - \mathbf{z})\|_1$ holds, where we have used that $\mathbf{x}$ is supported on S. Thus, we get

$$\|\mathbf{x}\|_1 < \|\mathbf{N}_{S^c}(\mathbf{x} - \mathbf{z})\|_1 + \|\mathbf{N}_S\mathbf{z}\|_1 = \| - \mathbf{N}_{S^c}\mathbf{z}\|_1 + \|\mathbf{N}_S\mathbf{z}\|_1 = \|\mathbf{N}_{S^c}\mathbf{z}\|_1 + \|\mathbf{N}_S\mathbf{z}\|_1 = \|\mathbf{z}\|_1,$$

where the last equality holds by Lemma 6.1.2. This completes the proof. ■

The above theorem clearly implies the following corollary, when we consider subsets $S \subseteq \{1, 2, \dots, n\}$ with $\mathrm{card}(S) \leq s$.

6.2.7 Corollary *Let $\boldsymbol{\Theta}$ be a wide matrix of a full rank and given a set $S \subseteq \{1, 2, \dots, n\}$. Then the following are equivalent*

(1) *If $\mathbf{x}$ is s-sparse, then $\mathbf{x}$ is the unique solution of the problem* $\arg\min\{\|\mathbf{s}\|_1 \mid \boldsymbol{\Theta}\mathbf{s} = \boldsymbol{\Theta}\mathbf{x}\}$.

(2) *The matrix $\boldsymbol{\Theta}$ has NSP of order s.*

Chapter 7

The Restricted Isometry Property

We have discussed that we can transform our ℓ_0 optimisation problem into the $(L1O)$ problem 5.1.1, which we know how to solve (more about this in Section 5.2). Of course, certain conditions must be satisfied for this transformation to be feasible, and in the previous Chapter 6, we introduced one such condition — the *Null Space Property* (NSP) of the ambient matrix $\mathbf{\Theta}$. Unfortunately, verifying whether a matrix satisfies NSP is not easy. Therefore, we will now introduce another property of matrices called the *Restricted Isometry Property* (RIP).[1] This property is also more robust against noise and it is popular in Compressed Sensing.

An isometry is a mapping that preserves distances. For RIP, we will not require strict isometry, but rather "almost isometry". We do not demand "almost isometry" for the entire matrix $\mathbf{\Theta}$ but only for individual submatrices and, moreover, we allow a certain deviation from the isometry requirement.

In order to be able to introduce the Restricted Isometry Property, we need to cover two auxiliary topics: in Section 7.1 we introduce *shrinking of vectors and matrices* and in Section 7.2 we provide certain additional facts about norms in $\mathbb{R}^n$ and we also introduce the operator norm. Finally, in Section 7.3 we prove the main result of this chapter: a matrix having RIP also has NSP, under certain conditions.

7.1 Shrinking vectors and matrices

Analogously to Section 6.1 we need a certain procedure pertaining to vectors before we will be able to delve into RIP. This time we want to "shrink" a vector $\mathbf{v}$ to a given prescribed non-empty set S of indices.

For example, given a vector $\begin{bmatrix} -3 \\ 2 \\ 1 \\ 4 \end{bmatrix}$ and a set $S = \{1, 3, 4\}$, we want to produce the vector $\begin{bmatrix} -3 \\ 1 \\ 4 \end{bmatrix}$ that has $\mathrm{card}(S)$ components which are copied from the original vector and which are prescribed by the set S.

Again, the above procedure is given by applying a certain linear map that we define now in full generality.

7.1.1 Definition Suppose $\emptyset \neq S = \{i_1, i_2, \ldots, i_{\mathrm{card}(S)}\} \subseteq \{1, 2, \ldots, n\}$ is given, where $i_1 < i_2 \ldots < i_{\mathrm{card}(S)}$. We denote by $\mathbf{P}_S$ the matrix with n columns and $\mathrm{card}(S)$ rows such that its k-th row is $\mathbf{e}_{i_k}^T$ for $k = 1, 2, \ldots, \mathrm{card}(S)$.

Thus, for example, for $S = \{1, 3, 4\} \subseteq \{1, 2, 3, 4\}$ we have

$$\mathbf{P}_S = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

and the equality

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} -3 \\ 2 \\ 1 \\ 4 \end{bmatrix} = \begin{bmatrix} -3 \\ 1 \\ 4 \end{bmatrix}$$

holds, as expected.

[1] The Restricted Isometry Property was introduced in [6], the textbook account is given in [15].

7.2 Additional facts about norms

In this section we gather further facts about ℓ_p norms on $\mathbb{R}^n$ that will be useful later. We also introduce the concept of an *operator norm* and we state (without proof) various equivalent ways how to express this norm.

7.2.1 Lemma *Given vector $\mathbf{x} \in \mathbb{R}^n$. Then the following inequalities*

$$\begin{aligned}\|\mathbf{x}\|_2 &\le \|\mathbf{x}\|_1 \le \sqrt{n}\|\mathbf{x}\|_2,\\ \|\mathbf{x}\|_\infty &\le \|\mathbf{x}\|_2 \le \sqrt{n}\|\mathbf{x}\|_\infty,\\ \|\mathbf{x}\|_\infty &\le \|\mathbf{x}\|_1 \le n\|\mathbf{x}\|_\infty.\end{aligned}$$

hold. In particular, we have the following chain of inequalities

$$\|\mathbf{x}\|_\infty \le \|\mathbf{x}\|_2 \le \|\mathbf{x}\|_1 \le \sqrt{n}\|\mathbf{x}\|_2 \le n\|\mathbf{x}\|_\infty.$$

Proof. For the proof, we will use definitions of ℓ_p norms and we separate individual inequalities. Below, the index i ranges from 1 to n.

(1) $\|\mathbf{x}\|_2^2 = \sum_i |x_i|^2 \le \sum_i |x_i|^2 + \sum_{i\neq j} |x_i||x_j| = \left(\sum_i |x_i|\right)^2 = \|\mathbf{x}\|_1^2$

(2) $\|\mathbf{x}\|_\infty^2 = \left(\max_i |x_i|\right)^2 \le \sum_i x_i^2 = \|\mathbf{x}\|_2^2$

(3) $\|\mathbf{x}\|_\infty = \max_i |x_i| \le \sum_i |x_i| = \|\mathbf{x}\|_1$

(4) $\|\mathbf{x}\|_1 = \sum_i |x_i| \le n \cdot \max_i |x_i| = n \cdot \|\mathbf{x}\|_\infty$

(5) $\|\mathbf{x}\|_2^2 = \sum_i x_i^2 \le n \cdot (\max |x_i|)^2 = n \cdot \|\mathbf{x}\|_\infty^2$

(6) Using the Cauchy-Bunyakovsky-Schwarz inequality for the standard scalar product, we obtain
$\|\mathbf{x}\|_1 = \sum_i |x_i| \cdot 1 \overset{C.S.B.}{\le} \left(\sum_i |x_i|^2\right)^{\frac{1}{2}} \cdot \left(\sum_i 1^2\right)^{\frac{1}{2}} = \|\mathbf{x}\|_2 \cdot \sqrt{n}$

■

7.2.2 Remark In case $\mathbf{x}$ is an s-sparse vector in $\mathbb{R}^n$, the inequality $\|\mathbf{x}\|_1 \le \sqrt{n}\|\mathbf{x}\|_2$ can be sharpened to

$$\|\mathbf{x}\|_1 \le \sqrt{s}\|\mathbf{x}\|_2.$$

The reasoning is analogous to that in the proof of Lemma 7.2.1.

7.2.3 Lemma *If vectors $\mathbf{u} \in \mathbb{R}^n$ and $\mathbf{v} \in \mathbb{R}^n$ satisfy*

$$\max_i |u_i| \le \min_j |v_j| \quad \text{for all } i, j = 1, 2, \ldots, n, \tag{7.1}$$

then

$$\|\mathbf{u}\|_2 \le \frac{1}{\sqrt{n}}\|\mathbf{v}\|_1.$$

Proof. From Lemma 7.2.1 and the definition of ℓ_p norms we have

$$\|\mathbf{u}\|_2 \le \sqrt{n}\|\mathbf{u}\|_\infty = \sqrt{n} \cdot \max |u_i|,$$

thus

$$\frac{\|\mathbf{u}\|_2}{\sqrt{n}} \le \max |u_i| \quad \text{for all } i = 1, 2, \ldots, n.$$

Then it obviously holds

$$\|\mathbf{v}\|_1 = \sum_j |v_j| \geq n \cdot \min |v_j|,$$

thus

$$\frac{\|\mathbf{v}\|_1}{n} \geq \min |v_j| \quad \text{for all } j = 1, 2, \ldots, n.$$

Using the initial assumption (7.1), we have

$$\frac{\|\mathbf{v}\|_1}{n} \geq \min |v_j| \geq \max |u_i| \geq \frac{\|\mathbf{u}\|_2}{\sqrt{n}}.$$

Then we obtain the final inequality

$$\|\mathbf{u}\|_2 \leq \frac{1}{\sqrt{n}} \|\mathbf{v}\|_1.$$

■

Any norm on $\mathbb{R}^n$ induces a norm on the vector space $Lin(\mathbb{R}^n, \mathbb{R}^n)$ of all linear maps from $\mathbb{R}^n$ to $\mathbb{R}^n$ (that can be identified with $n \times n$ matrices). This is the so-called operator norm that we define now.

7.2.4 Definition Let $\mathbf{T}$ be an $n \times n$ real matrix, let $\mathbf{z}$ be a vector in $\mathbb{R}^n$ and let $\| - \|$ be any norm on $\mathbb{R}^n$. We define the *operator norm* of $\mathbf{T}$ to be the number

$$\|\mathbf{T}\| = \sup_{\mathbf{z} \neq \mathbf{0}} \frac{\|\mathbf{T}\mathbf{z}\|}{\|\mathbf{z}\|}.$$

7.2.5 Remark Of course, we should prove that Definition 7.2.4 is correct, that is: we should prove that

(a) the supremum $\sup_{\mathbf{z} \neq \mathbf{0}} \frac{\|\mathbf{T}\mathbf{z}\|}{\|\mathbf{z}\|}$ exists

and

(b) the assignment $\mathbf{T} \mapsto \sup_{\mathbf{z} \neq \mathbf{0}} \frac{\|\mathbf{T}\mathbf{z}\|}{\|\mathbf{z}\|}$ indeed defines a norm in the sense of Definition 4.1.1.

We refer to [22] for the proof. In fact, we provide the following facts about the operator norms without proofs (which can be found in Appendix A of [15]).

(1) The following equalities

$$\|\mathbf{T}\| = \sup_{\mathbf{z} \neq \mathbf{0}} \frac{\|\mathbf{T}\mathbf{z}\|}{\|\mathbf{z}\|} = \sup_{\|\mathbf{y}\|=1} \|\mathbf{T}\mathbf{y}\| = \sup_{\|\mathbf{x}\| \leq 1} \|\mathbf{T}\mathbf{x}\|.$$

hold. Thus, the norm $\|\mathbf{T}\|$ of $\mathbf{T}$ can be expressed by any of the three expressions above.

(2) If we start with the ℓ_2 norm $\| - \|_2$ on $\mathbb{R}^n$ (i.e., the norm that is induced by the standard scalar product $\langle -, - \rangle$ on $\mathbb{R}^n$), then the induced operator norm is denoted by $\| - \|_{2 \to 2}$ and the equality

$$\|\mathbf{T}\|_{2 \to 2} = \sup_{\substack{\|\mathbf{x}\| \leq 1 \\ \|\mathbf{y}\| \leq 1}} |\langle \mathbf{T}\mathbf{x}, \mathbf{y} \rangle|$$

holds for any $n \times n$ matrix $\mathbf{T}$.

(3) If the matrix $\mathbf{T}$ is *symmetric*, then we have the equality

$$\|\mathbf{T}\|_{2 \to 2} = \sup_{\|\mathbf{x}\| \neq 0} \frac{|\langle \mathbf{T}\mathbf{x}, \mathbf{x} \rangle|}{\langle \mathbf{x}, \mathbf{x} \rangle}.$$

7.3 The Restricted Isometry Property

We formulate now the Restricted Isometry Property for matrices and we show that RIP implies NSP (under some additional conditions).

7.3.1 Definition Let $\mathbf{\Theta}$ be a wide matrix of a full rank.

(1) The s-th restricted isometry constant $\delta_s(\mathbf{\Theta})$ of $\mathbf{\Theta}$ is the smallest $\delta \geq 0$ such that

$$(1-\delta)\|\mathbf{x}\|_2 \leq \|\mathbf{\Theta x}\|_2^2 \leq (1+\delta)\|\mathbf{x}\|_2 \tag{7.2}$$

for all s-sparse vectors $\mathbf{x} \in \mathbb{R}^n$, where $s \in \{1, 2, \ldots, n\}$.

(2) We say that $\mathbf{\Theta}$ has RIP of order s if $\delta_s(\mathbf{\Theta}) < 1$.

We now show that the restricted isometry constant of a matrix can be expressed using the operator norm.

For the sake of readability, we write

$$\mathbf{v}_S$$

for the "shrunk" vector $\mathbf{P}_S \cdot \mathbf{v}$, and

$$\mathbf{\Theta}_S$$

for the "shrunk" submatrix $\mathbf{P}_S \cdot \mathbf{\Theta}$ (see Definition 7.1.1).

7.3.2 Proposition *The s-th restricted isometry constant $\delta_s(\mathbf{\Theta})$ of a wide matrix $\mathbf{\Theta}$ of a full rank is given by*

$$\delta_s(\mathbf{\Theta}) = \max_{\substack{S\subseteq\{1,2,\ldots,n\},\\ \mathrm{card}(S)\leq s}} \|\mathbf{\Theta}_S^T\mathbf{\Theta}_S - \mathbf{E}_{\mathrm{card}(S)}\|_{2\to 2}.$$

Proof. In the proof, for the sake of readability, we write

$$\|\mathbf{v}\| \qquad \|\mathbf{M}\| \qquad \mathbf{E}$$

for the ℓ_2 norm of a vector $\mathbf{v}$, for the induced operator norm of a matrix $\mathbf{M}$, and for the appropriate identity matrix, respectively.

For the proof, we will take the inequality (7.2) from Definition 7.3.1 (1) and rewrite it using $\langle \mathbf{x}, \mathbf{x}\rangle = \mathbf{x}^T\mathbf{x} = \|\mathbf{x}\|^2$ and using the fact that the vector $\mathbf{x}$ is s-sparse. We will thus focus only on vectors of the form $\mathbf{x}_S \in \mathbb{R}^{\mathrm{card}(S)}$ and the submatrix $\mathbf{\Theta}_S$, where $S \subseteq \{1, 2, ..., n\}$ and $\mathrm{card}(S) \leq s$. We obtain following equivalent chains of inequalities:

$$\begin{aligned}
(1-\delta)\langle \mathbf{x}_S, \mathbf{x}_S\rangle &\leq \langle \mathbf{\Theta}_S\mathbf{x}_S, \mathbf{\Theta}_S\mathbf{x}_S\rangle \leq (1+\delta)\langle \mathbf{x}_S, \mathbf{x}_S\rangle \\
(1-\delta)\mathbf{x}_S^T\mathbf{x}_S &\leq \mathbf{x}_S^T\mathbf{\Theta}_S^T\mathbf{\Theta}_S\mathbf{x}_S \leq (1+\delta)\mathbf{x}_S^T\mathbf{x}_S \\
-\delta\mathbf{x}_S^T\mathbf{x}_S &\leq \mathbf{x}_S^T(\mathbf{\Theta}_S^T\mathbf{\Theta}_S - \mathbf{E})\mathbf{x}_S \leq \delta\mathbf{x}_S^T\mathbf{x}_S \\
-\delta\langle \mathbf{x}_S, \mathbf{x}_S\rangle &\leq \langle \mathbf{x}_S, (\mathbf{\Theta}_S^T\mathbf{\Theta}_S - \mathbf{E})\mathbf{x}_S\rangle \leq \delta\langle \mathbf{x}_S, \mathbf{x}_S\rangle
\end{aligned}$$

$$\frac{|\langle \mathbf{x}_S|(\mathbf{\Theta}_S^T\mathbf{\Theta}_S - \mathbf{E})\mathbf{x}_S\rangle|}{\langle \mathbf{x}_S|\mathbf{x}_S\rangle} \leq \delta, \quad \mathbf{x_S} \neq \mathbf{o} \tag{7.3}$$

Observe that from this it follows

$$\max_{\substack{S\subseteq\{1,2,\ldots,n\},\\ \mathrm{card}(S)\leq s}} \quad \max_{\mathbf{x_S}\neq\mathbf{o}} \frac{|\langle \mathbf{x}_S|(\mathbf{\Theta}_S^T\mathbf{\Theta}_S - \mathbf{E})\mathbf{x}_S\rangle|}{\langle \mathbf{x}_S|\mathbf{x}_S\rangle} \leq \delta \tag{7.4}$$

and also that (7.4) implies (7.3) so these expressions are equivalent.

Now we will use the result 7.2.5 (3) and rewrite (7.4) using the operator norm:

$$\max_{\substack{S\subseteq\{1,2,\dots,n\},\\ \mathrm{card}(S)\leq s}} \|\mathbf{\Theta}_S^T\mathbf{\Theta}_S - \mathbf{E}\| \leq \delta.$$

Since $\delta_S(\mathbf{\Theta})$ is defined as the smallest such δ, we obtain the desired formula. ■

For proving our main result (Theorem 7.3.7 below), we need the following auxiliary result.

7.3.3 Lemma *Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ be s-sparse vectors with disjoint supports, i.e., $\mathrm{supp}(\mathbf{x}) \cap \mathrm{supp}(\mathbf{y}) = \emptyset$. Let $\mathbf{\Theta}$ be a wide matrix of a full rank. Then the inequality*

$$|\langle \mathbf{\Theta}\mathbf{x}, \mathbf{\Theta}\mathbf{y}\rangle| \leq \delta_{2s}(\mathbf{\Theta}) \cdot \|\mathbf{x}\|_2 \cdot \|\mathbf{y}\|_2.$$

holds.

Proof. We will use the same relaxation of notation as in the proof of Proposition 7.3.2.

Put $S = \mathrm{supp}(\mathbf{x}) \cup \mathrm{supp}(\mathbf{y})$. Since $\mathrm{supp}(\mathbf{x}) \cap \mathrm{supp}(\mathbf{y}) = \emptyset$, the equalities $\langle \mathbf{x}, \mathbf{y}\rangle = \langle \mathbf{x}_S, \mathbf{y}_S\rangle = 0$ hold. Therefore we can write

$$|\langle \mathbf{\Theta}\mathbf{x}, \mathbf{\Theta}\mathbf{y}\rangle| = |\langle \mathbf{\Theta}_S\mathbf{x}_S, \mathbf{\Theta}_S\mathbf{y}_S\rangle - \langle \mathbf{x}_S, \mathbf{y}_S\rangle| = |\mathbf{x}_S^T\mathbf{\Theta}_S^T\mathbf{\Theta}_S\mathbf{y}_S - \mathbf{x}_S^T\mathbf{y}_S| = |\mathbf{x}_S^T(\mathbf{\Theta}_S^T\mathbf{\Theta}_S - \mathbf{E})\mathbf{y}_S| = |\langle \mathbf{x}_S, (\mathbf{\Theta}_S^T\mathbf{\Theta}_S - \mathbf{E})\mathbf{y}_S\rangle|.$$

Using the Cauchy–Bunyakovsky–Schwarz inequality, we have

$$|\langle \mathbf{x}_S, (\mathbf{\Theta}_S^T\mathbf{\Theta}_S - \mathbf{E})\mathbf{y}_S\rangle| \leq \|\mathbf{x_S}\| \cdot \|(\mathbf{\Theta}_S^T\mathbf{\Theta}_S - \mathbf{E})\mathbf{y}_S\|.$$

And from properties of the operator norm, we obtain

$$\|\mathbf{x}_S\| \cdot \|(\mathbf{\Theta}_S^T\mathbf{\Theta}_S - \mathbf{E})\mathbf{y}_S\| \leq \|\mathbf{x}_S\| \cdot \|\mathbf{\Theta}_S^T\mathbf{\Theta}_S - \mathbf{E}\| \cdot \|\mathbf{y}_S\|.$$

Here is the final result:

$$|\langle \mathbf{\Theta}\mathbf{x}, \mathbf{\Theta}\mathbf{y}\rangle| \leq \|\mathbf{\Theta}_S^T\mathbf{\Theta}_S - \mathbf{E}\| \cdot \|\mathbf{x}_S\| = \delta_{2s}(\mathbf{\Theta}) \cdot \|\mathbf{x}\| \cdot \|\mathbf{y}\|.$$

■

Finally, we come to the statement that will give us an interesting result: RIP implies NSP, of course under some conditions on the ambient matrix. And as we already know from the previous chapter (specifically the Theorem 6.2.6), if a matrix has NSP it means we can transform our hard problem of ℓ_0 optimisation into the $(L1O)$ problem 5.1.1 that we can solve.

For the proof of Theorem 7.3.7 it will be convenient to introduce the following notation.

7.3.4 Notation Let $\mathbf{v}$ be a vector in $\mathbb{R}^n$ and let s be a natural number with $0 < s \leq n$. Let $n = q \cdot s + r$ be the division of n by s with the remainder $0 \leq r < s$. We denote by

$$S_0, S_1, \dots, S_{q-1}, S_q$$

the following sequence of subsets of $\{1, 2, \dots, n\}$:

(1) S_0 is the set of those s indices in $\{1, 2, \dots, n\}$ where the moduli of entries of $\mathbf{v}$ are the largest

(2) for any $0 < k \leq q - 1$

S_k is the set of those s indices in $(S_0 \cup \ldots \cup S_{k-1})^{\mathsf{c}}$ where the moduli of entries of $\mathbf{v}$ are the largest

(3) $S_q = (S_0 \cup \ldots \cup S_{q-1})^{\mathsf{c}}$

To illustrate how Notation 7.3.4 works, let us show the following example.

7.3.5 Example Let $\mathbf{v}$ $\begin{bmatrix}12\\9\\8\\11\\7\\6\\5\\4\\10\\3\\2\end{bmatrix}$ be a vector in $\mathbb{R}^{11}$ and let $s = 3$. Then $S_0 = \{1, 4, 9\}$, $S_1 = \{2, 3, 5\}$, $S_2 = \{6, 7, 8\}$, $S_3 = \{10, 11\}$.

Recall now Definition 6.1.1 of the S-nullifying matrix $\mathbf{N}_S$. For a vector $\mathbf{v} \in \mathbb{R}^n$ and $0 < s \leq n$, the following analogue of Lemma 6.1.2 holds.

7.3.6 Lemma *Let $\mathbf{v}$ be a vector in $\mathbb{R}^n$ and let $0 < s \leq n$. For the sequence $S_0, S_1, \ldots, S_q$ of sets defined in Notation 7.3.4 the following equality*

$$\mathbf{v} = \mathbf{N}_{S_0}\mathbf{v} + \mathbf{N}_{S_1}\mathbf{v} + \ldots + \mathbf{N}_{S_q}\mathbf{v}$$

holds.

Proof. Use the fact that S_0, $S_1, \ldots$, S_q is a partition of the set $\{1, 2, \ldots, n\}$ and then use Lemma 6.1.2. ■

7.3.7 Theorem *Let $\boldsymbol{\Theta}$ be a wide matrix of a full rank, let $0 < s \leq \frac{n}{2}$. If $\delta_{2s}(\boldsymbol{\Theta}) < \frac{1}{3}$, then $\boldsymbol{\Theta}$ has NSP of order s.*

Proof. For any vector $\mathbf{v} \in \ker(\boldsymbol{\Theta}) \setminus \{\mathbf{o}\}$ and the given s, construct the sequence $S_0, S_1, \ldots, S_q$ of subsets of $\{1, 2, \ldots, n\}$ as in Notation 7.3.4. We claim that it is sufficient to prove that the inequality

$$\|\mathbf{N}_{S_0}\mathbf{v}\|_2 \leq \frac{\delta_{2s}(\boldsymbol{\Theta})}{1 - \delta_{2s}(\boldsymbol{\Theta})} \cdot \frac{1}{\sqrt{s}}\|\mathbf{v}\|_1 \tag{7.5}$$

holds. The reason for this claim is divided into several steps:

(1) We claim that (7.5) implies the inequality

$$2\|\mathbf{N}_{S_0}\mathbf{v}\|_1 < \|\mathbf{v}\|_1.$$

For $\delta_{2s}(\boldsymbol{\Theta}) < \frac{1}{3}$ the inequality

$$\frac{\delta_{2s}(\boldsymbol{\Theta})}{1 - \delta_{2s}(\boldsymbol{\Theta})} < \frac{1}{2}$$

must hold. Hence, (7.5) implies the inequality

$$\|\mathbf{N}_{S_0}\mathbf{v}\|_2 \leq \frac{1}{2} \cdot \frac{1}{\sqrt{s}}\|\mathbf{v}\|_1$$

or, equivalently,

$$2 \cdot \sqrt{s}\|\mathbf{N}_{S_0}\mathbf{v}\|_2 \leq \|\mathbf{v}\|_1. \tag{7.6}$$

By Remark 7.2.2 the inequality

$$\|\mathbf{N}_{S_0}\mathbf{v}\|_1 \leq \sqrt{s} \cdot \|\mathbf{N}_{S_0}\mathbf{v}\|_2$$

holds. Combine that with (7.6) and obtain the desired inequality

$$2\|\mathbf{N}_{S_0}\mathbf{v}\|_1 < \|\mathbf{v}\|_1. \tag{7.7}$$

(2) Inequality 7.7 says that $\boldsymbol{\Theta}$ has NSP relative to set S_0. In fact, this is the statement of Proposition 6.2.4.

(3) Item (2) of this proof, the construction of S_0 from Notation 7.3.4 and Remark 6.2.3 (c) yields that $\mathbf{\Theta}$ has NSP of order s, as desired.

Thus, it remains to prove inequality (7.5).

Use Lemma 7.3.6 and the fact that $\mathbf{v} \in \ker(\mathbf{\Theta})$ to conclude that

$$\mathbf{o} = \mathbf{\Theta}\mathbf{v} = \mathbf{\Theta}(\mathbf{N}_{S_0}\mathbf{v} + \mathbf{N}_{S_1}\mathbf{v} + \dots \mathbf{N}_{S_q}\mathbf{v})$$

or, equivalently, that the equality

$$\mathbf{\Theta}\mathbf{N}_{S_0}\mathbf{v} = \mathbf{\Theta}(-\mathbf{N}_{S_1}\mathbf{v} - \dots - \mathbf{N}_{S_q}\mathbf{v}) \tag{7.8}$$

holds.

By assumption that matrix $\mathbf{\Theta}$ has RIP, the inequality

$$(1 - \delta_{2s}(\mathbf{\Theta})) \cdot \|\mathbf{N}_{S_0}\mathbf{v}\|_2^2 \leq \|\mathbf{\Theta}\mathbf{N}_{S_0}\mathbf{v}\|_2^2$$

or, equivalently, the inequality

$$\|\mathbf{N}_{S_0}\mathbf{v}\|_2^2 \leq \frac{1}{1 - \delta_{2s}(\mathbf{\Theta})} \cdot \|\mathbf{\Theta}\mathbf{N}_{S_0}\mathbf{v}\|_2^2 \tag{7.9}$$

must hold.

Now use (7.8) plus the fact that ℓ_2 norm is induced by the standard scalar product to rewrite the right-hand side of (7.9) as

$$\frac{1}{1 - \delta_{2s}(\mathbf{\Theta})} \cdot \langle \mathbf{\Theta}\mathbf{N}_{S_0}\mathbf{v}, \mathbf{\Theta}(-\sum_{k=1}^{q} \mathbf{N}_{S_k}\mathbf{v})\rangle = \frac{1}{1 - \delta_{2s}(\mathbf{\Theta})} \cdot \sum_{k=1}^{q} \langle \mathbf{\Theta}\mathbf{N}_{S_0}\mathbf{v}, \mathbf{\Theta}(-\mathbf{N}_{S_k}\mathbf{v})\rangle. \tag{7.10}$$

Since $\langle \mathbf{\Theta}\mathbf{N}_{S_0}\mathbf{v}, \mathbf{\Theta}(-\mathbf{N}_{S_k}\mathbf{v})\rangle \leq |\langle \mathbf{\Theta}\mathbf{N}_{S_0}\mathbf{v}, \mathbf{\Theta}\mathbf{N}_{S_k}\mathbf{v}\rangle|$ holds for every $k \in \{1, 2, \dots, q\}$, one can rewrite (7.9) using (7.10) to the inequality

$$\|\mathbf{N}_{S_0}\mathbf{v}\|_2^2 \leq \frac{1}{1 - \delta_{2s}(\mathbf{\Theta})} \cdot \sum_{k=1}^{q} |\langle \mathbf{\Theta}\mathbf{N}_{S_0}\mathbf{v}, \mathbf{\Theta}\mathbf{N}_{S_k}\mathbf{v}\rangle| \tag{7.11}$$

Observe that the pair $\mathbf{N}_{S_0}\mathbf{v}$ and $\mathbf{N}_{S_k}\mathbf{v}$ satisfy the assumptions of Lemma 7.3.3, for every $k \in \{1, 2, \dots, q\}$. Hence, by this lemma, inequality (7.11) yields

$$\|\mathbf{N}_{S_0}\mathbf{v}\|_2^2 \leq \frac{\delta_{2s}(\mathbf{\Theta})}{1 - \delta_{2s}(\mathbf{\Theta})} \cdot \sum_{k=1}^{q} \|\mathbf{N}_{S_0}\mathbf{v}\|_2 \cdot \|\mathbf{N}_{S_k}\mathbf{v}\|_2.$$

The above inequality can be divided by $\|\mathbf{N}_{S_0}\mathbf{v}\|_2$ (which is non-zero, since $\mathbf{v} \neq \mathbf{o}$). We obtain

$$\|\mathbf{N}_{S_0}\mathbf{v}\|_2 \leq \frac{\delta_{2s}(\mathbf{\Theta})}{1 - \delta_{2s}(\mathbf{\Theta})} \cdot \sum_{k=1}^{q} \|\mathbf{N}_{S_k}\mathbf{v}\|_2. \tag{7.12}$$

Due to the construction of the sequence $S_0, S_1, \dots, S_q$ in Notation 7.3.4, we have

$$\max_i |(\mathbf{N}_{S_k}\mathbf{v})_i| \leq \min_j |(\mathbf{N}_{S_{k-1}}\mathbf{v})_j|$$

for every $k \in \{1, 2, \dots, q\}$. Thus, Lemma 7.2.3 applies and we obtain the inequality

$$\|\mathbf{N}_{S_k}\mathbf{v}\|_2 \leq \frac{1}{\sqrt{s}}\|\mathbf{N}_{S_{k-1}}\mathbf{v}\|_1 \tag{7.13}$$

for every $k \in \{1, 2, \ldots, q\}$. Using (7.13) we can rewrite (7.12) and we obtain

$$\|\mathbf{N}_{S_0}\mathbf{v}\|_2 \leq \frac{\delta_{2s}(\boldsymbol{\Theta})}{1-\delta_{2s}\boldsymbol{\Theta}} \cdot \frac{1}{\sqrt{s}} \sum_{k=1}^{q} \|\mathbf{N}_{S_{k-1}}\mathbf{v}\|_1. \tag{7.14}$$

Since $\sum_{k=1}^{q} \|\mathbf{N}_{S_{k-1}}\mathbf{v}\|_1 \leq \sum_{k=1}^{q+1} \|\mathbf{N}_{S_{k-1}}\mathbf{v}\|_1 = \sum_{k=0}^{q} \|\mathbf{N}_{S_k}\mathbf{v}\|_1 = \|\mathbf{v}\|_1$ holds (the last equality holds by Lemma 7.3.6), the expression (7.14) implies that

$$\|\mathbf{N}_{S_0}\mathbf{v}\|_2 \leq \frac{\delta_{2s}(\boldsymbol{\Theta})}{1-\delta_{2s}(\boldsymbol{\Theta})} \cdot \frac{1}{\sqrt{s}} \|\mathbf{v}\|_1$$

holds, which is the desired inequality (7.5). ■

Let us now summarise what we have achieved in this chapter: we have demonstrated that matrices $\boldsymbol{\Theta}$ having RIP also have NSP (under the additional condition that the RIP-constant of $\boldsymbol{\Theta}$ is "sufficiently small"). This implies that our hard problem of the ℓ_0 minimisation can be transformed into a solvable problem of the ℓ_1 minimisation, as discussed in Chapter 5. Now, let us consider in some detail which matrices have RIP. That is the focus of the next chapter.

Chapter 8

The Restricted Isometry Property for random matrices

We have seen in Chapter 7 that any matrix $\mathbf{\Theta}$ having RIP with small restricted isometry constant has NSP. Consequently, as explained in Chapter 5, such a matrix $\mathbf{\Theta}$ will be a suitable matrix for the method of Compressed Sensing. A natural question now arises: do we have sufficiently many examples of matrices that satisfy RIP? The answer is quite peculiar:

There are plenty of RIP matrices but it is rather difficult to produce an explicit example of such a matrix.

The above statement is no paradox: most known constructions of RIP matrices make use of rather involved techniques of probability theory.

In this chapter we only indicate that matrices having a certain *random* pattern do indeed satisfy RIP. The full proofs of relevant results are out of the scope of this text. Thus, in Section 8.1 we collect the basic notions of Probability Theory and in Section 8.2 we briefly recall the Fourier transform. These steps are needed to formulate Theorem 8.3.4 below in Section 8.3.

8.1 A quick overview of probability theory

Probability theory is a large branch of mathematics with applications in many fields. In this thesis, we will only sketch some key concepts required for our purposes. We will not provide formal definitions, everything will be said only informally, so for example we will not delve into defining the probability, probability space etc. Details can be found, e.g., in [14].

As we have said, we will focus on randomness in the concept of RIP. Here are the informal definitions of the crucial concepts:

(1) A *probability space* is a set of events that we want to study. In formal probability theory this is a so-called *measurable space* i.e., a set M (of events) together with specified subsets of M (the "allowed" sets of events) that have to satisfy further axioms.

(2) A *random variable* is (usually) a "well-behaved" function X from a probability space to (a subset of) real numbers. The random variable X can either be *continuous* or *discrete*. Continuous random variables take values in a certain interval of the reals, whereas the discrete random variables take values in a (finite) discrete set of real numbers. A good example of a continuous random variable can be measured temperatures, of a discrete variable when you roll a dice.

(3) A *cumulative distribution function* F of a random variable X evaluated at x is the probability that X will take a value less than or equal to x. It is the function $F : \mathbb{R} \to [0, 1]$ defined by

$$F(x) = P(X \leq x) \quad \text{for } -\infty < x < \infty$$

(4) For a continuous random variable X, we introduce a *probability density function* f, which represents "a relative likelihood that the value of X will be equal to x". The density function f of X is any function

such that $f(x) \geq 0$ for all x, $\int_{-\infty}^{\infty} f(x)\,dx = 1$ and such that the equality

$$P(a \leq X \leq b) = \int_a^b f(x)\,dx$$

holds for any a, b with $a \leq b$.

(5) An *expected value* of a random variable X, denoted by $\mathbb{E}X$, generalises the weighted average. For a discrete random variable X taking values $x_1, x_2, \ldots$ with probabilities $P(X = x_i) = p_i$, the expected value is defined by

$$\mathbb{E}X = \sum_i x_i \cdot p_i,$$

if this series converges.

For a continuous random variable X with probability density function $f(x)$, the expected value is defined by

$$\mathbb{E}X = \int_{-\infty}^{\infty} x \cdot f(x)\,dx,$$

if this integral exists.

(6) A *variance* of a random variable X is the expected value of the squared deviation from the mean of X. It is defined by

$$\text{var } X = \mathbb{E}(X - \mathbb{E}X)^2.$$

There are usually some "rules" telling us how the events are likely to behave. Thus, several fundamental models of probability distributions exist, and we now list some of them. We indicate special forms for the expected value, variance, cumulative distribution function and probability density function.

(1) *Bernoulli distribution* $Ber(p)$ is an example for a discrete random variable which takes the value 1 with probability p and the value 0 with probability $1-p$. The example of the random variable with the Bernoulli distribution is the number of ones that come up when rolling a dice.

$$F(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 - p & \text{for } 0 \leq x < 1 \\ 1 & \text{for } x \geq 1 \end{cases}$$

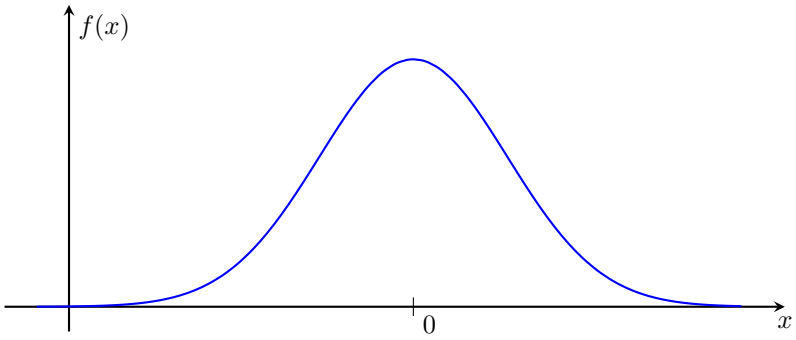
$$\mathbb{E}X = p, \quad \text{var } X = p(1-p)$$

(2) *Gaussian (Normal) distribution* $N(\mu, \sigma^2)$ is a type of continuous probability distribution. This distribution is very important because a lot of random variables has it (it is a good model for example for the height of individuals in the population).

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \quad -\infty < x < \infty$$
$$F(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{x} e^{-\frac{(t-\mu)^2}{2\sigma^2}}\,dt, \quad -\infty < x < \infty$$

$$\mathbb{E}X = \mu, \quad \text{var } X = \sigma^2$$

Below you can see a graph illustrating the probability density function of a Gaussian distribution.



8.2 A brief insight into the Fourier transform

In this section we very informally discuss the Fourier transform. The Fourier transform is a powerful tool used almost everywhere; we have already mentioned it shortly in Section 2.1. It decomposes a function into sines and cosines and it is used for converting signals from the time domain to the frequency domain (thus we obtain the so-called *spectrum* of the signal). To convert spectrum back to the time domain we use the inverse Fourier transform.

First, let us consider periodic signals. A "well-behaved" periodic function s with period $T > 0$ can be decomposed into a sum of sines and cosines[1] as follows:, we use the so-called *Fourier series* (FS) defined by

$$s(t) = \sum_{n=-\infty}^{\infty} c_n e^{\frac{2\pi i n t}{T}},$$

where c_n are the Fourier coefficients given by

$$c_n = \frac{1}{T} \int_a^{a+T} s(t) e^{-\frac{2\pi i n t}{T}} dt, \quad a \in \mathbb{R}.$$

Thus, from a periodic continuous signal, we obtain its spectrum which will be discrete and non-periodic. In the case of a discrete periodic signal, we would proceed very similarly, with the integral replaced by a sum, and now we would use the so-called *Discrete Fourier series* (DFS). In this case, the spectrum would also be discrete, but periodic.

Now, we would like to generalise the Fourier series for non-periodic signals. The idea is as follows: we extend the period of the periodic signal to infinity in the limit. Here, the information about certain frequencies will no longer suffice, we will need information about all frequencies, and the obtained spectrum in the case of non-periodic signals will thus no longer be discrete but continuous.

The *Fourier transform* (FT) of a "well-behaved" function s is defined by

$$S(\omega) = \int_{-\infty}^{\infty} s(t) e^{-i\omega t} dt, \quad \omega \in \mathbb{R}, \quad \omega = 2\pi f,$$

and the *inverse Fourier transform* is defined by

$$s(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} S(\omega) e^{i\omega t} d\omega.$$

Thus, from a non-periodic continuous signal, we obtain its spectrum which will be continuous and non-periodic. Again, there is also a tool for discrete non-periodic signals, the so-called *Discrete-time Fourier transform*. Their spectrum would be continuous and periodic.

Moreover, there is the so-called *Discrete Fourier transform*, which complements the above four tools (FS, DFS, FT, DtFT) for computing the spectrum. It allows for the numerical computation of the spectrum from a sample of the signal, thus, we do not need to know the signal on the entire domain, however, we can obtain only an approximate result. DFT is a part of almost all computer algebra programs and it is most commonly

[1] The formulas are simpler to write down if one uses the complex exponential $e^{it} = \cos t + i \sin t$. The passage to sines and cosines can be done by taking the imaginary and the real part, respectively.

implemented as the so-called *Fast Fourier transform* (FFT), which is an efficient numerical algorithm for computing DFT.

The Discrete Fourier transform transforms a finite sequence of complex numbers $\{x_n\} = (x_0, x_1, ..., x_{N-1})$ into another finite sequence of complex numbers $\{X_k\} = (X_0, X_1, ..., X_{N-1})$, where

$$X_k = \sum_{n=0}^{N-1} x_n \cdot e^{\frac{-2i\pi kn}{N}}, \quad k = 0, 1, ..., N-1.$$

Above, we would not delve into technicalities of Fourier transform and Fourier series. For details, we refer, e.g., to [19].

8.3 The Restricted Isometry Property theorem for random matrices

We will formulate a theorem stating that random matrices have RIP *with high probability*.

8.3.1 Definition We say that matrix is a *random matrix* if it has random variables as its entries.

8.3.2 Example A random matrix $\mathbf{\Theta}$ with independent and identically distributed standard Gaussian random variables as its entries is called *Gaussian random matrix*. The following matrix is Gaussian [20].

$$\mathbf{G} = \begin{bmatrix} 1.2448 & 0.0561 & -0.8778 & 1.1058 & 1.1759 & 0.7339 \\ -0.1854 & 0.7819 & -1.3124 & 0.8786 & 0.3965 & -0.3138 \\ -0.4925 & -0.6234 & 0.0307 & 0.8448 & -0.2629 & 0.7013 \\ 0.1933 & -1.5660 & 2.3387 & 0.4320 & -0.0535 & 0.2294 \\ -1.0143 & -0.7578 & 0.3923 & 0.3935 & -0.4883 & -2.7609 \\ -1.8839 & 0.4546 & -0.4495 & 0.0972 & -2.6562 & 1.3405 \end{bmatrix}$$

8.3.3 Example Here is the example of the Fourier matrix.

$$\mathbf{F} = \begin{bmatrix} 1.0000 + 0.0000i & 1.0000 + 0.0000i & 1.0000 + 0.0000i & 1.0000 + 0.0000i & 1.0000 + 0.0000i \\ 1.0000 + 0.0000i & 0.3090 - 0.9511i & -0.8090 - 0.5878i & -0.8090 + 0.5878i & 0.3090 + 0.9511i \\ 1.0000 + 0.0000i & -0.8090 - 0.5878i & 0.3090 + 0.9511i & 0.3090 - 0.9511i & -0.8090 + 0.5878i \\ 1.0000 + 0.0000i & -0.8090 + 0.5878i & 0.3090 - 0.9511i & 0.3090 + 0.9511i & -0.8090 - 0.5878i \\ 1.0000 + 0.0000i & 0.3090 + 0.9511i & -0.8090 + 0.5878i & -0.8090 - 0.5878i & 0.3090 - 0.9511i \end{bmatrix}$$

We now state the result saying that for any wide random matrix $\mathbf{\Theta}$ one can claim *with high probability* that $\mathbf{\Theta}$ satisfies RIP of order s, provided that $\mathbf{\Theta}$ has "sufficiently many" rows. More in detail, the following holds.[2]

8.3.4 Theorem *Let $m < n$ be positive integers, let $\delta > 0$ and let $s = O(\frac{m}{\log^4 n})$.*[3] *Let $\mathbf{\Theta}$ be the random matrix with n columns and m rows defined by one of the following methods.*

(1) *(Normal entries) Let the entries of $\mathbf{\Theta}$ be independent and identically distributed with a normal distribution $N(0, 1/m)$.*

(2) *(Bernoulli entries) Let the entries of $\mathbf{\Theta}$ be independent and identically distributed with a Bernoulli distribution taking the values $\pm 1/\sqrt{m}$, each with 50% probability.*

(3) *(Random rows of the Discrete Fourier transform) Let $A \subseteq \{0, \ldots, n-1\}$ be a random subset of size m. Let $\mathbf{\Theta}$ be the matrix obtained from the Discrete Fourier transform matrix (i.e. the matrix F with entries $F[\ell, j] = e^{-2\pi i \ell j/n}/\sqrt{n}$ for $\ell, j \in \{0, \ldots, n-1\}$) by selecting the rows indexed by A, where i is the imaginary unit.*

Then $\mathbf{\Theta}$ has RIP of order s with $\delta_s(\mathbf{\Theta}) = \delta$ with probability $p \approx 1 - e^{-n}$.

[2] In item (3) of Theorem 8.3.4, we consider matrices with *complex entries*. Although we have not developed the theory of Compressed Sensing for complex matrices in this text, such a theory, of course, exists. See, for example, [15].

[3] We use the "Big O" notation from Computer Science. The equality $s = O(\frac{m}{\log^4 n})$ means that, roughly, s is "asymptotically equal" to $\frac{m}{\log^4 n}$.

The proof is beyond the scope of this work, we will not provide it here, we refer to [2], [28]. The theory behind random matrices having RIP is very complex and would require an entire separate work; we have only outlined that matrices having RIP exist and what such matrices look like. Now, we should have all the necessary information to understand some basic aspects of Compressed Sensing. Since important facts have been mentioned throughout all chapters, the final summary that follows offers an overview of our key findings, as far as Compressed Sensing in MRI is concerned.

Summary

In this thesis, we introduced the concept of Compressed Sensing, a relatively new method from 2006 that significantly improves many aspects of our lives. One example of its application is magnetic resonance imaging. Magnetic resonance scan could be very discomforting for patients. But thanks to Compressed Sensing, the examination can last for a shorter duration of time. Let us now recapitulate the problem we formulated in this work and how we dealt with its solution.

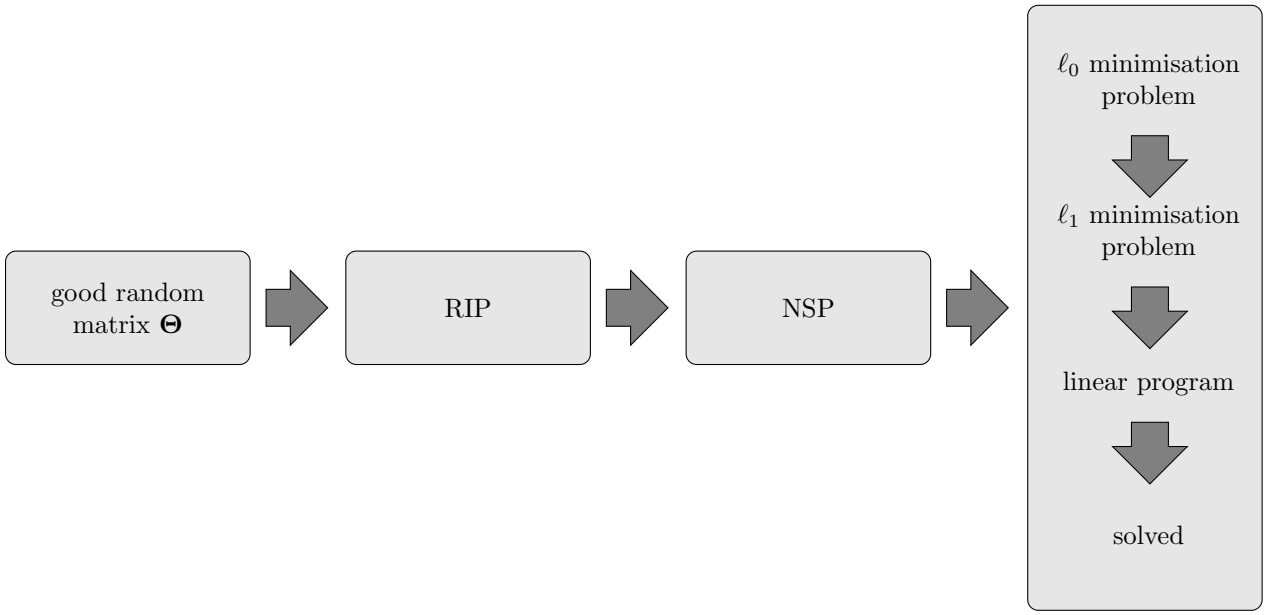
We have a vector of measured data $\mathbf{y}$ and a matrix $\mathbf{\Theta}$ and we are looking for $\mathbf{s}$ that is as sparse as possible while satisfying the equation $\mathbf{\Theta s} = \mathbf{y}$, i.e.,

$$\text{Given wide matrix } \mathbf{\Theta} \text{ with full rank and vector } \mathbf{y}. \text{ Find } \arg\min\{\|\mathbf{s}\|_0 \mid \mathbf{\Theta s} = \mathbf{y},\ \mathbf{s} \in \mathbb{R}^n\},$$

which was formulated in Chapter 5 as an instance of the (*SAS*) problem 4.4.1. We showed that this ℓ_0 minimisation problem is *NP*-hard and we cannot, therefore, solve it efficiently. However, under certain conditions, we can replace (*SAS*) with the ℓ_1 minimisation problem 5.1.1, which can be written as a linear program and efficiently solved. To be able to do that, our matrix $\mathbf{\Theta}$ must be *special*. Namely, in Chapter 6, we introduced the Null Space Property (NSP) that enables this transformation. Unfortunately, it is not easy to verify whether $\mathbf{\Theta}$ has NSP. Therefore, we introduced the Restricted Isometry Property (RIP) in Chapter 7, which implies NSP under certain conditions. Constructing a matrix having RIP is difficult, but thankfully, with the use of randomness, we can obtain such matrices.

Now let us recapitulate it all once more, this time in reverse. It will be stated very informally; for all the implications mentioned, certain conditions, which were discussed in previous chapters, are necessary .

We have a *good* random matrix $\mathbf{\Theta}$. It has RIP, which means it also has NSP. Because it has NSP, we can transform the ℓ_0 minimisation problem into the ℓ_1 minimisation problem. That we write as a linear program and solve it.



The Compressed Sensing method is indeed used in magnetic resonance imaging, the software enabling Compressed Sensing is provided to MR devices, for example, by companies such as Philips [26] and Siemens [11].

Index

algorithm, 15

Basis Pursuit, 29
Bernoulli distribution, 47

compression, 11, 12
cumulative distribution function, 46

decision problem, 16, 26

expected value, 47

Fourier matrix, 49
Fourier series, 48
Fourier transform, 12, 48

Gaussian distribution, 47
Gaussian matrix, 49

ℓ_p norm, 20, 29, 39
ℓ_∞ norm, 21
linear program, 31

magnetic resonance, 8, 12

norm, 20, 29, 39
normal distribution, 47
NP problem, 17
NP-complete problem, 18
NP-hard problem, 18, 26
Null Space Property, 33
Nyquist–Shannon Sampling Theorem, 13

operator norm, 39
optimisation problem, 14, 18, 24, 29

P problem, 16
partial derivative, 22
probability density function, 46
probability space, 46
problem
    (L1LP), 32
    (L1O), 29
    (L2O), 24
    (LP), 31
    (SAS), 26
    (SAS-decision), 26
    (TOP), 18
    (TOP-decision), 19
    (X3C), 26
pseudoinverse, 25

random matrix, 49
random variable, 46
restricted isometry constant, 41
Restricted Isometry Property, 38

sparse, 22
support, 21

total derivative, 23

variance, 47

wide matrix, 24

Bibliography

[1] Robert A. Adams and Christopher Essex, *Calculus: A Complete Course* (10th ed.), Addison Wesley, 2021. 22

[2] Richard Baraniuk, Mark Davenport, Ronald DeVore, and Michael Wakin, A simple proof of the restricted isometry property for random matrices, *Constructive Approximation*, Vol. 58, No. 3 (2008), 253–263. 50

[3] Adi Ben-Israel and Thomas N.E. Greville, *Generalised Inverses* (2nd ed.), Springer, 2003. 25

[4] Stephen Boyd and Lieven Vandenberghe, *Convex Optimization* (2nd ed.), Cambridge University Press, 2004. 18

[5] Steven L. Brunton and J. Nathan Kutz, *Data-driven Science and Engineering*, Cambridge University Press, 2019. 14

[6] Emmanuel Candès and Terence Tao, Decoding by Linear Programming, *IEEE Transactions on Information Theory*, Vol. 51, No. 12 (2005), 4203–4215. 38

[7] Henri Cartan, *Differential Calculus*, Hermann and Houghton-Mifflin Company, 1971. 22

[8] Edwin K. P. Chong and Stanisław H. Żak, *An Introduction to Optimization* (4th ed.), John Wiley & Sous, 2013. 25

[9] Alonzo Church, An Unsolvable Problem of Elementary Number Theory, *American Journal of Mathematics*, Vol. 58, No. 2. (1936), 345–363. 15

[10] Albert Cohen, Wolfgang Dahmen and Ronald DeVore, Compressed Sensing and Best k-term Approximation, *Journal of the American Mathematical Society*, Vol. 22 (2009), 211–231. 33

[11] Compressed Sensing Space, Siemens Healthineers, `https://www.siemens-healthineers.com/en-uk/magnetic-resonance-imaging/options-and-upgrades/clinical-applications/compressed-sensing-space`, accessed on 14 May, 2024. 51

[12] Stephen A. Cook, *The Complexity of Theorem-Proving Procedures*, ACM SIGACT, 1971, 151–158. 18

[13] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, *Introduction to Algorithms* (3rd ed), MIT Press, 2009. 16

[14] Frederik M. Dekking, Cornelis Kraaikamp, Hendrik P. Lopuhaä and Ludolf E. Meester, *A Modern Introduction to Probability and Statistics*, Springer, 2005. 46

[15] Simon Foucart and Holger Rauhut, *A Mathematical Introduction to Compressive Sensing*, Birkhäuser, 2012. 13, 33, 38, 40, 49

[16] Michael R. Garey and David S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness* (3rd ed.), Cambridge University Press, 2017. 15, 18, 26

[17] Frederick S. Hillier and Gerald J. Lieberman, *Introduction to Operations Research* (7th ed), McGraw-Hill, 2001. 31

[18] John E. Hopcroft, Rajeev Motwani and Jeffrey D. Ullman, *Introduction to Automata Theory, Languages and Computation* (3rd ed), Pearson, 2006. 16

[19] David W. Kammler, *A First Course in Fourier Analysis*, Cambridge University Press, 2007. 49

[20] Giacomo Livan, Marcel Novaes and Pierpaolo Vivo, *Introduction to Random Matrices*, Springer, 2018. 49

[21] Donald W. McRobbie, Elizabeth A. Moore and Martin J. Graves and Martin R. Prince, *MRI from Picture to Proton* (3rd ed), Cambridge University Press, 2017. 8, 10

[22] Carl D. Meyer, *Matrix analysis and applied linear algebra* (2nd ed.), SIAM, 2023. 40

[23] Balas K. Natarajan, Sparse Approximate Solutions to Linear Systems *SIAM Journal on Computing*, Vol. 24, No. 2 (1995), 227–234. 26

[24] Harry Nyquist, Certain Topics in Telegraph Transmission Theory, *Transactions of the AIEE*, Vol. 47, No. 2 (1928), 617–644. 13

[25] Obituary of Raymond Vahan Damadian, Armenian American Health Professionals Organization, https://aahpo.org/in-memoriam-raymond-vahan-damadian/, accessed on 14 May, 2024. 9

[26] Philips Compressed SENSE, Philips MRI Systems & Solutions, https://www.philips.co.uk/healthcare/resources/landing/the-next-mr-wave/compressed-sense, accessed on 14 May, 2024. 51

[27] Irina Rish and Genady Ya. Grabarnik, *Sparse Modeling*, CRC Press, 2015. 13

[28] Mark Rudelson and Roman Vershynin, On Sparse Reconstruction from Fourier and Gaussian Measurements *Communication on Pure and Applies Mathematics*, Vol. 61, No. 8 (2008), 1025–1045. 50

[29] C. E. Shannon, Communication in the Presence of Noise, *Proceedings of the IRE*, Vol. 37, No. 1 (1949), 10–21. 13

[30] Eric Schechter, *Handbook of Analysis and Its Foundations*, Academic Press, 1997. 21

[31] Peter Smith, *An Introduction to Gödel's Theorems*, Cambridge University Press, 2013. 15

[32] Daniel A. Spielman and Shang-Hua Teng, Smoothed Analysis of Algorithms: Why the Simplex Algorithm Usually Takes Polynomial Time, *Journal of the ACM*, Vol. 51, No. 3 (2004), 385–463. 31

[33] Gilbert Strang, *Linear Algebra and its Applications* (4th ed.) Belmont,Thomson Brooks/Cole, 2006. 20, 25

[34] Alan M. Turing, On Computable Numbers, with an Application to the Entscheidungsproblem, *Proceedings of the London Mathematical Society*, Vol. 2, No. 42 (1937), 230–265. 15

[35] Alan M. Turing, Computability and λ-definability, *The Journal of Symbolic Logic*, Vol. 2, No. 4 (1937), 153–163. 15

[36] Catherine Westbrook and Carolyn Kaut, *MRI in Practice*, Blackwell Science, 1998. 10

[37] Wikipedia page of Isidor Isaac Raabi, https://en.wikipedia.org/wiki/Isidor_Isaac_Rabi, accessed on 14 May, 2024. 9

[38] Wikipedia page of Felix Bloch, https://en.wikipedia.org/wiki/Felix_Bloch, accessed on 14 May, 2024. 9

[39] Wikipedia page of Edward Mills Purcell, https://en.wikipedia.org/wiki/Edward_Mills_Purcell, accessed on 9 December, 2023. 9

[40] Wikipedia page of Paul Lauterbur, https://en.wikipedia.org/wiki/Paul_Lauterbur, accessed on 9 December, 2023. 9

[41] Wikipedia page of Peter Mansfield, https://en.wikipedia.org/wiki/Peter_Mansfield, accessed on 14 May, 2024. 9

[42] Wikipedia page of Alan Turing, https://en.wikipedia.org/wiki/Alan_Turing, accessed on 14 May, 2024. 16

[43] Wikipedia page of Stephen Cook, https://en.wikipedia.org/wiki/Stephen_Cook, accessed on 14 May, 2024. 16