1 Introduction to Information Theory and Data Science

Miguel R. D. Rodrigues*, Stark C. Draper∓, Waheed U. Bajwa†, and Yonina C. Eldar‡

*University College London, UK, ∓University of Toronto, Canada

†Rutgers University–New Brunswick, USA, ‡Weizmann Institute of Science, Israel

The field of information theory—dating back to 1948—is one of the landmark intellectual achievements of the 20th century. It provides the philosophical and mathematical underpinnings of the technologies that allow accurate representation, efficient compression, and reliable communication of sources of data. A wide range of storage and transmission infrastructure technologies, including optical and wireless communication networks, the internet, and audio and video compression, have been enabled by principles illuminated by information theory. Technological breakthroughs based on information-theoretic concepts have driven the "information revolution" characterized by the anywhere and anytime availability of massive amounts of data and fueled by the ubiquitous presence of devices that can capture, store, and communicate data.

The existence and accessibility of such massive amounts of data promise immense opportunities, but also pose new challenges in terms of how to extract useful and actionable knowledge from such data streams. Emerging data science problems are different from classical ones associated with the transmission or compression of information in which the semantics of the data is unimportant. That said, we are starting to see that information-theoretic methods and perspectives can, in a new guise, play important roles in understanding emerging data-science problems. The goal of this book is to explore such new roles for information theory and to understand better the modern interaction of information theory with other data-oriented fields such as statistics and machine learning.

The purpose of this chapter is to set the stage for the book and for the upcoming chapters. We first overview classical information-theoretic problems and solutions. We then discuss emerging applications of information-theoretic methods in various data science problems and, where applicable, refer the reader to related chapters in the book. Throughout this chapter, we highlight the perspectives, tools, and methods that play important roles in classic information-theoretic paradigms and in emerging areas of data science. Table 1.1 provides a summary of the different topics covered in this chapter and highlights the different chapters that can be read as a follow up to these topics.

Table 1.1 Major topics covered in this chapter and their connections to other chapters.

Section(s)	Topic	Related Chapter(s)
1.1–1.4	An Introduction to Information Theory	15
1.6	Information Theory and Data Acquisition	2–4, 6, 16
1.7	Information Theory and Data Representation	5, 11
1.8	Information Theory and Data Analysis/Processing	6–16

1.1 Classical Information Theory: A Primer

Claude Shannon's 1948 paper "A Mathematical Theory of Communications," *Bell Systems Technical Journal*, July/Oct. 1948, laid out a complete architecture for digital communication systems [1]. In addition, it articulated the philosophical decisions for the design choices made. **Information theory**, as Shannon's framework has come to be known, is a beautiful and elegant example of engineering science. It is all the more impressive as Shannon presented his framework decades before the first digital communication system was implemented, and at a time when digital computers were in their infancy.

Figure 1.1 presents a general schematic of a digital communication system. This figure is a reproduction of Shannon's "Figure 1" from his seminal paper. Before 1948 no one had conceived of a communication system in this way. Today nearly all digital communication systems obey this structure.

The flow of information through the system is as follows. An **information source** first produces a random message that a transmitter wants to convey to a destination. The message could be a word, a sentence or a picture. In information theory all information sources are modeled as being sampled from a set

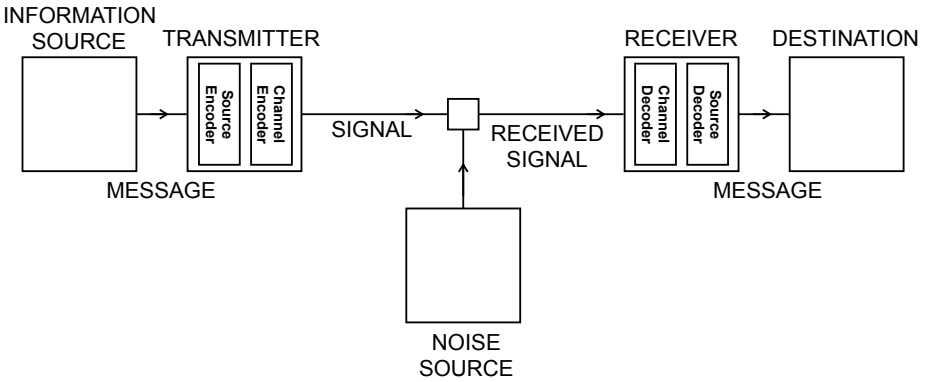

Figure 1.1 Reproduction of Shannon's Figure 1 in [1] with the addition of the source and channel encoding/decoding blocks. In Shannon's words, this is a "Schematic diagram of a general communication system."

of possibilities according to some probability distribution. Modeling information sources as stochastic is a key aspect of Shannon's approach. It allowed him to quantify uncertainty as the lack of knowledge and reduction in uncertainty as the gaining of knowledge or "information."

The message is then fed into a transmission system. The transmitter itself has two main sub-components: the **source encoder** and the **channel encoder**. The source encoder converts the message into a sequence of 0's and 1's, i.e., a bit sequence. There are two classes of source encoders. **Lossless** source coding removes predictable redundancy that can later be recreated. In contrast, **lossy** source coding is an irreversible process wherein some distortion is incurred in the compression process. Lossless source coding is often referred to as **data compression** while lossy coding is often referred to as **rate-distortion** coding. Naturally, the higher the distortion the fewer the number of bits required.

The bit sequence forms the data payload that is fed into a channel encoder. The output of the channel encoder is a signal that is transmitted over a noisy communication medium. The purpose of the **channel code** is to convert the bits into a set of possible signals or **codewords** that can be reliably recovered from the noisy received signal.

The communication medium itself is referred to as the **channel**. The channel can model the physical separation of the transmitter and receiver. It can also, as in data storage, model separation in time.

The destination observes a signal that is the output of the communication channel. Similar to the transmitter, the receiver has two main components: a **channel decoder** and a **source decoder**. The former maps the received signal into a bit sequence that is, hopefully, the same as the bit sequence produced by the transmitter. The latter then maps the estimated bit sequence to an estimate of the original message.

If lossless compression is used then an apt **measure of performance** is the probability the message estimate at the destination is not equal to the original message at the transmitter. If lossy compression (rate-distortion) is used, then other measures of goodness, such as mean-squared error, are more appropriate.

Interesting questions addressed by information theory include the following:

1. Architectures
   - What tradeoffs in performance are incurred by the use of the architecture detailed in Figure 1.1?
   - When can this architecture be improved upon; when can it not?
2. Source coding: Lossless data compression
   - How should the information source be modeled; as stochastic, as arbitrary but unknown, or in some other way?
   - What is the shortest bit sequence into which a given information source can be compressed?
   - What assumptions does the compressor work under?
   - What are basic compression techniques?

3. Source coding: Rate-distortion theory
   - How do you convert an analog source into a digital bit stream?
   - How do you reconstruct/estimate the original source from the bit stream?
   - What is the trade-off involved between the number of bits used to describe a source and the distortion incurred in reconstruction of the source?
4. Channel coding
   - How should communication channels be modeled?
   - What throughput, measured in bits per second, at what reliability, measured in terms of probability of error, can be achieved?
   - Can we quantify fundamental limits on the realizable tradeoffs between throughput and reliability for a given channel model?
   - How does one build computationally tractable channel coding systems that "saturate" the fundamental limits?
5. Multi-user information theory
   - How do we design systems that involve multiple transmitters and receivers?
   - How do many (perhaps correlated) information sources and transmission channels interact?

The decades since Shannon's first paper have seen fundamental advances in each of these areas. They have also witnessed information-theoretic perspectives and thinking impacting a number of other fields including security, quantum computing and communications, and cryptography. The basic theory and many of these developments are documented in a cadre of excellent texts, including [2, 3, 4, 5, 6, 7, 8, 9]. Some recent advances in network information theory, which involves multiple sources and/or multiple destinations, are also surveyed in Chapter 15. In the next three sections we illustrate the basics of information-theoretic thinking by focusing on simple (point-to-point) binary sources and channels. In Sec. 1.2 we discuss the compression of binary sources. In Sec. 1.3 we discuss channel coding over binary channels. Finally, in Sec. 1.4, we discuss computational issues, focusing on linear codes.

1.2 Source Coding: Near-lossless Compression of Binary Sources

To gain a feel for the tools and results of classical information theory consider the following lossless source coding problem. One observes a length-n string of random coin flips, $X_1, X_2, \ldots, X_n$, each $X_i \in \{\text{heads}, \text{tails}\}$. The flips are independent and identically distributed with $\mathbb{P}(X_i = \text{heads}) = p$ where $0 \leq p \leq 1$ is a known parameter. Suppose we want to map this string into a bit sequence to store on a computer for later retrieval. Say we are going to assign a *fixed* amount of memory to store the sequence. How much memory must we allocate?

Since there are 2^n possible sequences, all of which could occur if p is not equal to 0 or 1, if we use n bits we can be 100% certain we could index any heads/tails sequence that we might observe. However, certain sequences, while

possible, are much less likely than others. Information theory exploits such non-uniformity to develop systems that can trade off between efficiency (the storage of fewer bits) and reliability (the greater certainty that one will later be able to reconstruct the observed sequence). In the following we accept some (arbitrarily) small probability $\epsilon > 0$ of observing a sequence that we choose not to be able to store a description of.[1] One can think of ϵ as the probability of the system failing. Under this assumption we derive bounds on the number of bits that need to be stored.

1.2.1 Achievability: An upper bound on the rate required for reliable data storage

To figure out which sequences we may choose not to store, let us think about the statistics. In expectation we observe np heads. Of the 2^n possible heads/tails sequences there are $\binom{n}{np}$ sequences with np heads. (For the moment we ignore non-integer effects and deal with them later.) There will be some variability about this mean but, at a minimum, we must be able to store all these expected realizations since these realizations all have the same probability. While $\binom{n}{np}$ is the cardinality of the set, we prefer to develop a good approximation that is more amenable to manipulation. Further, rather than counting cardinality, we will count the log-cardinality. This is because given k bits we can index 2^k heads/tails source sequences. Hence, it is the exponent in which we are interested.

Using Stirling's approximation to the factorial, $\log_2 n! = n\log_2 n - (\log_2 e)n + O(\log_2 n)$, and ignoring the order term we have

$$\log \binom{n}{np} \simeq n\log_2 n - n(1-p)\log_2(n(1-p)) - np\log_2 np \tag{1.1}$$

$$= n\log_2\left(\frac{1}{1-p}\right) + np\log_2\left(\frac{1-p}{p}\right)$$

$$= n\left[(1-p)\log_2\left(\frac{1}{1-p}\right) + p\log_2\left(\frac{1}{p}\right)\right]. \tag{1.2}$$

In (1.1) the $(\log_2 e)n$ terms have canceled and the term in square brackets in (1.2) is called the **(binary) entropy** which we denote as $H_B(p)$, so

$$H_B(p) = -p\log_2 p - (1-p)\log_2(1-p) \tag{1.3}$$

where $0 \le p \le 1$ and $0\log 0 = 0$. The binary entropy function is plotted in Fig. 1.2 within Sec. 1.3. One can compute that when $p = 0$ or $p = 1$ then $H_B(0) = H_B(1) = 0$. The interpretation is that, since there is only one all-tails and one all-heads sequence, and we are quantifying log-cardinality, there is only one sequence to index in each case so $\log_2(1) = 0$. In these cases we apriori know

[1] In source coding this is termed **near-lossless** source coding as the arbitrarily small ϵ bounds the probability of system failure and thus loss of the original data. In the **variable-length** source coding paradigm one stores a variable amount of bits per sequence, and minimizes the expected number of bits stored. We focus on the near-lossless paradigm as the concepts involved more closely parallel those in channel coding.

the outcome (respectively all heads or all tails) and so do not need to store any bits to describe the realization. On the other hand, if the coin is fair then $p = 0.5$, $H_B(0.5) = 1$, $\binom{n}{n/2} \simeq 2^n$, and we must use n bits of storage. In other words, on an exponential scale almost all binary sequences are 50% heads and 50% tails. As an intermediate value, if $p = 0.11$ then $H_B(0.11) \simeq 0.5$.

The operational upshot of (1.2) is that if one allocates $nH_B(p)$ bits then basically all expected sequences can be indexed. Of course, there are caveats. First, np may not be integer. Second, there will be variability about the mean. To deal with both, we allocate a few more bits, $n(H_B(p) + \delta)$ in total. We use these bits not just to index the expected sequences, but also the **typical sequences**, those sequences with empirical entropy close to the entropy of the source.[2] In the case of coin flips, if a particular sequence consists of n_H heads (and $n - n_H$ tails) then we say that the sequence is "typical" if

$$H_B(p) - \delta \leq \left[\frac{n_H}{n} \log_2 \left(\frac{1}{p} \right) + \frac{n - n_H}{n} \log_2 \left(\frac{1}{1-p} \right) \right] \leq H_B(p) + \delta. \tag{1.4}$$

It can be shown that the cardinality of the set of sequences that satisfies condition (1.4) is upper bounded by $2^{n(H_B(p)+\delta)}$. Therefore if, for instance, one lists the typical sequences lexiographically, then any typical sequence can be described using $n(H_B(p) + \delta)$ bits. One can also show that for any $\delta > 0$ the probability of the source *not* producing a typical sequence can be upper bounded by any $\epsilon > 0$ as n grows large. This follows from the law of large numbers. As n grows the distribution of the fraction of heads in the realized source sequence concentrates about its expectation. Therefore, as long as n is sufficiently large, and as long as $\delta > 0$, any $\epsilon > 0$ will do. The quantity $H_B(p) + \delta$ is termed the **storage "rate"** R. For this example $R = H_B(p) + \delta$. The rate is the amount of memory that must be made available per source symbol. In this case there were n symbols (n coin tosses) so one normalizes $n(H_B(p) + \delta)$ by n to get the rate $H_B(p) + \delta$.

The above idea can be immediately extended to independent and identically distributed (i.i.d.) finite-alphabet (and to more general) sources as well. The general definition of the **entropy of a finite-alphabet random variable** X with probability mass function (p.m.f.) p_X is

$$H(X) = -\sum_{x \in \mathcal{X}} p_X(x) \log_2 p_X(x), \tag{1.5}$$

where "finite-alphabet" means the sample space $\mathcal{X}$ is finite.

Regardless of the distribution (binary, non-binary, even non-i.i.d.), the simple coin-flipping example illustrates one of the central tenets of information theory. This is to focus one's design on what is likely to happen, i.e., the typical events, rather than on worst-case events. The partition of events into typical and atypical is, in information theory, known as the **asymptotic equipartition property** or

[2] In the literature these are termed the "weakly" typical sequences. There are other definitions of typicality that differ in terms of their mathematical use. The overarching concept is the same.

AEP. In a nutshell the simplest form of the AEP says that for long i.i.d. sequences one can, up to some arbitrarily small probability ϵ, partition all possible outcomes into two sets: the typical set and the atypical set. The probability of observing an event in the typical set is at least $1-\epsilon$. Furthermore, on an exponential scale all typical sequences are of equal probability. Designing for typical events is a hallmark of information theory.

1.2.2 Converse: A lower bound on the rate required for reliable data storage

A second hallmark of information theory is the emphasis on developing bounds. The source coding scheme described above is known as an **achievability result**. Achievability results involve describing an operational system that can, in principle, be realized in practice. Such results provide **(inner) bounds** on what is possible. The performance of the best system is at least this good. In the above example we developed a source coding technique that delivers high-reliability storage and requires a rate of $H(X)+\delta$ where both the error ϵ and the slack δ can be arbitrarily small if n is sufficiently large.

An important coupled question is how much (or whether) we can reduce the rate further, thereby improving the efficiency of the scheme. In information theory, **outer bounds** on what is possible—e.g., showing that if the encoding rate is too small one cannot guarantee a target level of reliability—are termed **converse results**.

One of the key lemmas used in converse results is **Fano's Inequality** [7], named for Robert Fano. The statement of the inequality is as follows: For any pair of random variables $(U,V) \in \mathcal{U} \times \mathcal{V}$ jointly distributed according to $p_{U,V}(\cdot,\cdot)$ and for any estimator $G : \mathcal{U} \to \mathcal{V}$ with probability of error, $P_e = \Pr[G(U) \neq V]$,

$$H(V|U) \leq H_B(P_e) + P_e \log_2(|\mathcal{V}|-1). \tag{1.6}$$

On the left-hand-side of (1.6) we encounter the **conditional entropy** $H(V|U)$ of the joint p.m.f. $p_{U,V}(\cdot,\cdot)$. We use the notation $H(V|U=u)$ to denote the entropy in V when the realization of the random variable U is set to $U=u$. Let us name this the "pointwise" conditional entropy, the value of which can be computed by applying our formula for entropy (1.5) to the p.m.f. $p_{V|U}(\cdot|u)$. The conditional entropy is the expected pointwise conditional entropy:

$$H(V|U) = \sum_{u\in\mathcal{U}} p_U(u) H(V|U=u) = \sum_{u\in\mathcal{U}} p_U(u) \left[\sum_{v\in\mathcal{V}} p_{V|U}(v|u) \log_2 \frac{1}{p_{V|U}(v|u)} \right]. \tag{1.7}$$

Fano's Inequality (1.6) can be interpreted as a bound on the ability of any hypothesis test function G to make a (single) correct guess of the realization of V based on its observation of U. As the desired error probability $P_e \to 0$, both terms on the right-hand-side go to zero, implying that the conditional entropy must be small. Conversely, if the left-hand-side is not too small, that asserts a non-zero lower bound on P_e. A simple explicit bound is achieved by

upper bounding $H_B(P_e)$ as $H_B(P_e) \leq 1$ and rearranging to find that $P_e \geq (H(V|U) - 1)/\log_2(|\mathcal{V}| - 1)$.

The usefulness of Fano's Inequality stems, in part, from the weak assumptions it makes. One can apply Fano's to any joint distribution. Often identification of an applicable joint distribution is part of the creativity in the use of Fano's. For instance in the source coding example above, one takes V to be the stored data sequence, so $|\mathcal{V}| = 2^{n(H_B(p)+\delta)}$, and U to be the original source sequence, i.e., $U = X^n$. While we do not provide the derivation herein, the result is that to achieve an error probability of at most P_e the storage rate R is lower bounded by $R \geq H(X) - P_e \log_2 |\mathcal{X}| - H_B(P_e)/n$ where $|\mathcal{X}|$ is the source alphabet size; for the binary example $|\mathcal{X}| = 2$. As we let $P_e \to 0$ we see that the lower bound on the achievable rate is $H(X)$ which, letting $\delta \to 0$, is also our upper bound. Hence we have developed an operational approach to data compression where the rate we achieve matches the converse bound.

We now discuss the interaction between achievability and converse results. As long as the compression rate $R > H(X)$ then due to concentration in measure, in the achievability case the failure probability $\epsilon > 0$ and rate slack $\delta > 0$ can both be chosen to be arbitrarily small. Concentration of measure occurs as the **blocklength** n becomes large. In parallel with n getting large the total number of bits stored nR also grows.

The entropy $H(X)$ thus specifies a boundary between two regimes of operation. When the rate R is larger than $H(X)$, achievability results tell us that arbitrarily reliable storage is possible. When R is smaller than $H(X)$, converse results imply that reliable storage is not possible. In particular, rearranging the converse expression and once again noting that $H_B(P_e) \leq 1$, the error probability can be lower bounded as

$$P_e \geq \frac{H(X) - R - 1/n}{\log_2 |\mathcal{X}|}. \tag{1.8}$$

If $R < H(X)$ then for n sufficiently large P_e is bounded away from zero.

The entropy $H(X)$ thus characterizes a **phase transition** between one state, the possibility of reliable data storage, and another, the impossibility. Such sharp information-theoretic phase transitions also characterize classical information-theoretic results on data transmission which we discuss in the next section, and applications of information-theoretic tools in the data sciences which we turn to later in the chapter.

1.3 Channel Coding: Transmission over the Binary-Symmetric Channel

Shannon applied the same mix of ideas (typicality, entropy, conditional entropy) to solve the, perhaps at first seemingly quite distinct, problem of reliable and effi-

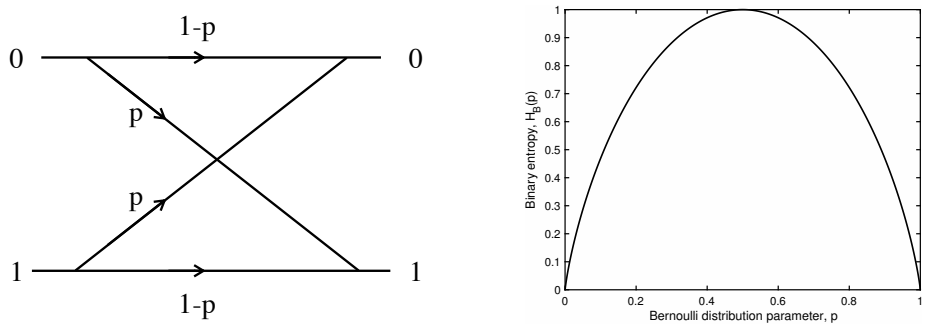

Figure 1.2 On the left we present a graphical description of the binary symmetric channel (BSC). Each transmitted binary symbol is represented as a 0 or 1 input on the left. Each received binary observation is represented by a 0 or 1 output on the right. The stochastic relationship between inputs and outputs is represented by the connectivity of the graph where the probability of transitioning each edge is represented by the edge label p or $1-p$. The channel is "symmetric" due to the symmetries in these transition probabilities. On the right we plot the binary entropy function $H_B(p)$ as a function of p, $0 \leq p \leq 1$. The capacity of the BSC is $C_{\text{BSC}} = 1 - H_B(p)$.

cient digital communications. This is typically referred to as Shannon's "channel coding" problem in contrast to the "source coding" problem already discussed.

To gain a sense of the problem we return to the simple binary setting. Suppose our source coding system has yielded a length-k string of "information bits." For simplicity we assume these bits are randomly distributed as before, i.i.d. along the sequence, but are now fair; i.e., each is equally likely to be "0" or a "1." The objective is to convey this sequence over a communications channel to a friend. Importantly we note that since the bits are uniformly distributed, our results on source coding tells us that no further compression is possible. Thus, uniformity of message bits is a worst-case assumption.

The channel we consider is the **binary-symmetric channel** or BSC. We can transmit binary symbols over the BSC. Each input symbol is conveyed to the destination, but not entirely accurately. The binary-symmetric channel "flips" each channel input symbol ($0 \to 1$ or $1 \to 0$) with probability p, $0 \leq p \leq 1$. Flips occur independently. The challenge is for the destination to deduce, hopefully with high accuracy, the k information bits transmitted. Due to the symbol flipping noise, we get some slack; we transmit $n \geq k$ binary channel symbols. For efficiency's sake, we want n to be as close to k as possible, while meeting the requirement of high reliability. The ratio k/n is termed the "rate" of communication. The length-n binary sequence transmitted is termed the "codeword." This "bit-flipping" channel can be used, e.g., to model data storage errors in a computer memory. A graphical representation of the BSC is depicted in Fig. 1.2.

1.3.1 Achievability: A lower bound on the rate of reliable data communication

The idea of channel coding is analogous to human-evolved language. The length-k string of information bits is analogous to what we think, i.e., the concept we want to get across to the destination. The length-n codeword string of binary channel symbols is analogous to what we say (the sentence). There is redundancy in spoken language that makes it possible for spoken language to be understood in noisy (albeit not too noisy) situations. We analogously engineer redundancy into what a computer transmits to be able to combat the expected (the typical!) noise events. For the BSC those would be the expected bit-flip sequences.

We now consider the noise process. For any chosen length-n codeword there are about $\binom{n}{np}$ typical noise patterns which, using the same logic as in our discussion of source compression, is a set of roughly $2^{nH_B(p)}$ patterns. If we call X^n the codeword and E^n the noise sequence then what the receiver measures is $Y^n = X^n + E^n$. Here addition is vector addition over $\mathbb{F}_2$, i.e., coordinate-wise where the addition of two binary symbols is implemented using the XOR operator. The problem faced by the receiver is to identify the transmitted codeword. One can imagine that if the possible codewords are far apart in the sense that they differ in many entries (i.e., their **Hamming distance** is large) then the receiver will be less likely to make an error when deciding on the transmitted codeword. Once such a codeword estimate is made it can then be mapped back to the length-k information bit sequence. A natural decoding rule, in fact the **maximum-likelihood rule**, is for the decoder to pick the codeword closest to Y^n in terms of Hamming distance.

The design of the codebook (analogous to the choice of grammatically correct and—thus allowable—sentences in a spoken language) is a type of probabilistic packing problem. The question is, how do we select the set of codewords so that the probability of a decoding error is small? We can develop a simple upper bound on how large the set of reliably-decodable codewords can be. There are 2^n possible binary output sequences. For any codeword selected there are $2^{nH_B(p)}$ typical output sequences, each associated with a typical noise sequence, that form a **noise ball** centered on the codeword. If we were simply able to divide up the output space into disjoint sets of cardinality $2^{nH_B(p)}$ we would end up with $2^n/2^{nH_B(p)} = 2^{n(1-H_B(p))}$ distinct sets. This **sphere packing** argument tells us that the best we could hope to do would be to transmit this number of distinct codewords reliably. Thus, the number of information bits k would equal $n(1 - H_B(p))$, i.e., a transmission rate of $1 - H_B(p)$.

Perhaps quite surprisingly, as n gets large, $1 - H_B(p)$ is the supremum of achievable rates at (arbitrarily) high reliability. This is the Shannon capacity $C_{\text{BSC}} = 1 - H_B(p)$. The result follows from the law-of-large numbers which can be used to show that the typical noise balls concentrate. Shannon's proof that one can actually find a configuration of codewords while keeping the probability of decoding error small was an early use of the **probabilistic method**. For any rate $R = C_{\text{BSC}} - \delta$, where $\delta > 0$ is arbitrarily small, a randomized choice of the

positioning of each codeword will with high probability yield a code with a small probability of decoding error. To see the plausibility of this statement we revisit the sphere packing argument. At rate $R = C_{\mathrm{BSC}} - \delta$ the 2^{nR} codewords are each associated with a typical noise ball of $2^{nH_B(p)}$ sequences. If the noise balls were all disjoint this would be a total of $2^{nR}2^{nH_B(p)} = 2^{n(1-H_B(p)-\delta)+nH_B(p)} = 2^{n(1-\delta)}$ sequences. As there are 2^n binary sequences, the fraction of the output space taken up by the union of typical noise spheres associated with the codewords is $2^{n(1-\delta)}/2^n = 2^{-n\delta}$. So, for any $\delta > 0$ fixed, as the blocklength $n \to \infty$, only an exponentially disappearing fraction of the output space is taken up by the noise balls. By choosing the codewords independently at random, each uniformly chosen over all length-n binary sequences, one can show that the expected (over the choice of codewords and channel noise realization) average probability of error is small. Hence, at least one codebook exists that performs at least as well as this expectation.

While Shannon showed the existence of such a code (actually a sequence of codes as $n \to \infty$), it took another half-century for researchers in error-correction coding to find asymptotically optimal code designs and associated decoding algorithms that were computationally tractable and therefore implementable in practice. We discuss this computational problem and some of these recent code designs in Sec. 1.4.

While the above example is set in the context of a binary-input and binary-output channel model, the result is a prototype of the results that hold for **discrete memoryless channels**. A discrete memoryless channel is described by the conditional distribution $p_{Y|X} : \mathcal{X} \to \mathcal{Y}$. Memoryless means that output symbols are conditionally independent given the input codeword, i.e., $p_{Y^n|X^n}(y^n|x^n) = \prod_{i=1}^n p_{Y|X}(y_i|x_i)$. The supremum of achievable rates is the **Shannon capacity** C where

$$C = \sup_{p_X}[H(Y) - H(Y|X)] = \sup_{p_X} I(X;Y). \tag{1.9}$$

In (1.9), $H(Y)$ is the entropy of the output space, induced by the choice of **input distribution** p_X via $p_Y(y) = \sum_{x\in\mathcal{X}} p_X(x)p_{Y|X}(y|x)$, and $H(Y|X)$ is the conditional entropy of $p_X(\cdot)p_{Y|X}(\cdot|\cdot)$. For the BSC the optimal choice of $p_X(\cdot)$ is uniform. We shortly develop an operational intuition for this choice by connecting it to hypothesis testing. We note that this choice induces the uniform distribution on Y. Since $|\mathcal{Y}| = 2$, this means that $H(Y) = 1$. Further, plugging the channel law of the BSC into (1.7) yields $H(Y|X) = H_B(p)$. Putting the pieces together recovers the Shannon capacity result for the binary symmetric channel, $C_{\mathrm{BSC}} = 1 - H_B(p)$.

In (1.9) we introduce the equality $H(Y) - H(Y|X) = I(X;Y)$ where $I(X;Y)$ denotes the **mutual information** of the joint distribution $p_X(\cdot)p_{Y|X}(\cdot|\cdot)$. The mutual information is another name for the **Kullback-Leibler (KL) divergence** between the joint distribution $p_X(\cdot)p_{Y|X}(\cdot|\cdot)$ and the product of the joint distribution's marginals, $p_X(\cdot)p_Y(\cdot)$. The general formula for the KL divergence

between a pair of distributions p_U and p_V defined over a common alphabet $\mathcal{A}$ is

$$D(p_U \| p_V) = \sum_{a \in \mathcal{A}} p_U(a) \log_2 \frac{p_U(a)}{p_V(a)}. \tag{1.10}$$

In the definition of mutual information over $\mathcal{A} = \mathcal{X} \times \mathcal{Y}$, $p_{X,Y}(\cdot, \cdot)$ plays the role of $p_U(\cdot)$ and $p_X(\cdot)p_Y(\cdot)$ plays the role of $p_V(\cdot)$.

The KL divergence arises in hypothesis testing where it is used to quantify the error exponent of a binary hypothesis test. Conceiving of channel decoding as a hypothesis testing problem—which one of the codewords was transmitted?—helps us understand why (1.9) is the formula for the Shannon capacity. One way the decoder can make its decision regarding the identity of the true codeword is to test each codeword against independence. In other words, does the empirical joint distribution of any particular codeword X^n and the received data sequence Y^n look jointly distributed according to the channel law or does it look independent? That is, does (X^n, Y^n) look like it is distributed i.i.d. according to $p_{XY}(\cdot, \cdot) = p_X(\cdot)p_{Y|X}(\cdot|\cdot)$ or i.i.d. according to $p_X(\cdot)p_Y(\cdot)$? The exponent of the error in this test is $-D(p_{XY} \| p_X p_Y) = -I(X;Y)$. Picking the input distribution p_X to maximize (1.9) maximizes this exponent. Finally, via an application of the union bound we can assert that, roughly, $2^{nI(X;Y)}$ codewords can be allowed before more than one codeword in the codebook appear to be jointly distributed with the observation vector Y^n according to p_{XY}.

1.3.2 Converse: An upper bound on the rate of reliable data communication

An application of Fano's Inequality (1.6) shows that C is also an upper bound on the achievable communication rate. This application of Fano's Inequality is similar to that used in source coding. In this application of (1.6) we set $V = X^n$ and $U = Y^n$. The greatest additional subtlety is that we must leverage the memoryless property of the channel to **single-letterize** the bound. To single-letterize means to express the final bound in terms of only the $p_X(\cdot)p_{Y|X}(\cdot|\cdot)$ distribution, rather than in terms of the joint distribution of the length-n input and output sequences. This is an important step because n is allowed to grow without bound. By single-letterizing we express the bound in terms of a fixed distribution, thereby making the bound computable.

As at the end of the discussion of source coding, in channel coding we find a boundary between two regimes of operation: the regime of efficient and reliable data transmission, and the regime where such reliable transmission is impossible. In this instance, the phase transition boundary is marked by the Shannon capacity C.

1.4 Linear Channel Coding

In the previous sections we discussed the sharp phase transitions in both source and channel coding discovered by Shannon. These phase transitions demarc fundamental boundaries between what is possible and what is not. In practice one desires schemes that "saturate" these bounds. In the case of source coding we can saturate the bound if we can design source coding techniques with rates that can be made arbitrarily close to $H(X)$ (from above). For channel coding we desire coding methods with rates that can be made arbitrarily close to C (from below). While Shannon discovered and quantified the bounds, he did not specify realizable schemes that attained them.

Decades of effort have gone into developing methods of source and channel coding. For lossless compression of memoryless sources, as in our motivating examples, good approaches such as Huffman and arithmetic coding were found rather quickly. On the other hand, finding computationally tractable and therefore implementable schemes of error correction coding that got close to capacity took much longer. For a long time it was not even clear that computational tractable techniques of error correction that saturated Shannon's bounds were even possible. For many years researchers thought that there might be a second phase transition at the **cuttoff rate**, only below which computationally tractable methods of reliable data transmission existed. (See [10] for a nice discussion.) Indeed only with the emergence of modern coding theory in the 1990s and 2000s that studies turbo, low-density parity-check (LDPC), spatially coupled LDPC, and Polar codes has the research community, even for the BSC, developed computationally tractable methods of error correction that closed the gap to Shannon's bound.

In this section we introduce the reader to linear codes. Almost all codes in use have linear structure, structure that can be exploited to reduce the complexity of the decoding process. As in the previous sections we only scratch the surface of the discipline of error-correction coding. We point the reader to the many excellent texts on the subject, e.g., [6, 11, 12, 13, 14, 15].

1.4.1 Linear codes and syndrome decoding

Linear codes are defined over finite fields. As we have been discussing the BSC, the field we will focus on is $\mathbb{F}_2$. The set of codewords of a length-n binary linear code correspond to a subspace of the vector space $\mathbb{F}_2^n$. To encode we use a matrix-vector multiplication defined over $\mathbb{F}_2$ to map a length-k column vector $\mathbf{b} \in \mathbb{F}_2^k$ of "information bits" into a length-n column vector $\mathbf{x} \in \mathbb{F}_2^n$ of binary "channel symbols" as

$$\mathbf{x} = \mathbf{G}^T \mathbf{b}, \tag{1.11}$$

where $\mathbf{G} \in \mathbb{F}_2^{k \times n}$ is a k by n binary "generator" matrix and $\mathbf{G}^T$ denotes the transpose of $\mathbf{G}$. Assuming that $\mathbf{G}$ is full rank, all 2^k possible binary vectors $\mathbf{b}$

are mapped by $\mathbf{G}$ into 2^k distinct codewords $\mathbf{x}$, so the set of possible codewords (the "**codebook**") is the row-space of $\mathbf{G}$. We compute the rate of the code as $R = k/n$.

Per our earlier discussion the channel adds the length-n noise sequence $\mathbf{e}$ to $\mathbf{x}$, yielding the channel output $\mathbf{y} = \mathbf{x} + \mathbf{e}$. To decode, the receiver pre-multiplies $\mathbf{y}$ by the **parity-check matrix** $\mathbf{H} \in \mathbb{F}_2^{m \times n}$ to produce the length-m **syndrome** $\mathbf{s}$ as

$$\mathbf{s} = \mathbf{H}\mathbf{y}. \tag{1.12}$$

We caution the reader not to confuse the parity check matrix $\mathbf{H}$ with the entropy function $H(\cdot)$. By design, the rows of $\mathbf{H}$ are all orthogonal to the rows of $\mathbf{G}$ and thus span the null-space of $\mathbf{G}$.[3] When the columns of $\mathbf{G}$ are linearly independent, the dimension of the nullspace of $\mathbf{G}$ is $n - k$ and the relation $m = n - k$ holds.

Substituting in the definition for $\mathbf{x}$ into the expression for $\mathbf{y}$ and thence into (1.12) we compute

$$\mathbf{s} = \mathbf{H}(\mathbf{G}^T\mathbf{b} + \mathbf{e}) = \mathbf{H}\mathbf{G}^T\mathbf{b} + \mathbf{H}\mathbf{e} = \mathbf{H}\mathbf{e}, \tag{1.13}$$

where the last step follows because the rows of $\mathbf{G}$ and $\mathbf{H}$ are orthogonal by design so that $\mathbf{H}\mathbf{G}^T = \mathbf{0}$, the $m \times k$ all-zeros matrix. Inspecting (1.13) we observe that the computation of the syndrome $\mathbf{s}$ yields m linear constraints on the noise vector $\mathbf{e}$.

Since $\mathbf{e}$ is of length n and $m = n - k$, (1.13) specifies an under-determined set of linear equations in $\mathbb{F}_2$. However, as already discussed, while $\mathbf{e}$ *could* be any vector, when the blocklength n becomes large, concentration of measure comes into play. With high probability the realization of $\mathbf{e}$ will concentrate around those sequences that contain only np non-zero elements. We recall that $p \in [0, 1]$ is the bit-flip probability and note that in $\mathbb{F}_2$ any non-zero must be a one. In coding theory we are therefore faced with the problem of solving an under-determined set of linear equations subject to a *sparsity constraint*: there are only about np non-zero elements in the solution vector. In fact, as fewer bit flips are more likely, the maximum likelihood solution for the noise vector $\mathbf{e}$ is to find the maximally-sparse vector that satisfies the syndrome constraints, i.e.,

$$\hat{\mathbf{e}} = \arg\min_{\mathbf{e} \in \mathbb{F}_2^n} d_H(\mathbf{e}) \quad \text{such that} \quad \mathbf{s} = \mathbf{H}\mathbf{e}, \tag{1.14}$$

where $d_H(\cdot)$ is the Hamming weight (or distance from 0^n) of the argument. As mentioned before, the Hamming weight is the number of non-zero entries of $\mathbf{e}$. It plays a role analogous to the cardinality function in $\mathbb{R}^n$ (sometimes denoted $\|\cdot\|_0$), often used to enforce sparsity in the solution to optimization problems.

We observe that there are roughly $2^{nH_B(p)}$ typical binary bit-flip sequences with roughly np non-zeros each. The syndrome $\mathbf{s}$ provides m linear constraints on the noise sequence. Each constraint is binary so that if all constraints are

[3] Note that in finite fields vectors can be self-orthogonal; e.g., in $\mathbb{F}_2$ any even-weight vector is orthogonal to itself.

linearly independent, each constraint reduces by 50% the set of possible noise sequences. Thus, if the number m of constraints exceeds $\log_2(2^{nH_B(p)}) = nH_B(p)$ we should be able to decode.[4]

Decoders can thus be thought of as solving a binary search problem where the measurements/queries are fixed ahead of time, and the decoder uses the results of the queries, often in an iterative fashion, to determine $\mathbf{e}$. Once $\hat{\mathbf{e}}$ is calculated, the codeword estimate $\hat{\mathbf{x}} = \mathbf{y} + \hat{\mathbf{e}} = \mathbf{G}^T\mathbf{b} + (\mathbf{e} + \hat{\mathbf{e}})$. If $\hat{\mathbf{e}} = \mathbf{e}$ then the term in brackets cancels and $\mathbf{b}$ can uniquely and correctly be recovered from $\mathbf{G}^T\mathbf{b}$. This last point follows since the codebook is the row-space of $\mathbf{G}$ and $\mathbf{G}$ is full rank.

Noting from the previous section that the capacity of the BSC channel is $C = 1 - H_B(p)$ and the rate of the code is $R = k/n$, we would achieve capacity if $1 - H_B(p) = k/n$ or, equivalently, if the syndrome length $m = n - k = n(1 - k/n) = nH_B(p)$. This is the objective of coding theory: to find "good" codes (specified by their generator matrix $\mathbf{G}$ or, alternately, by their parity-check matrix $\mathbf{H}$) and associated decoding algorithms (that attempt to solve (1.14) in a computationally efficient manner) so as to be able to keep $R = k/n$ as close as possible to $C_{\text{BSC}} = 1 - H_B(p)$.

1.4.2 From linear to computational tractable: Polar codes

To understand the challenge of designing computationally tractable codes say that, in the previous discussion, one picked $\mathbf{G}$ (or $\mathbf{H}$) according to the Bernoulli-0.5 random i.i.d. measure. Then for any fixed rate R if one sets the blocklength n to be sufficiently large the generator (or parity-check) matrix produced will, with arbitrarily high probability, specify a code that is capacity-achieving. Such a selection of $\mathbf{G}$ or $\mathbf{H}$ is respectively referred to as the Elias or the Gallager ensembles.

However, attempting to use the above capacity-achieving scheme can be problematic from a computational viewpoint. To see the issue consider block length $n = 4000$ and rate $R = 0.5$, which are well within normal ranges for these parameters. For these choices there are $2^{nR} = 2^{2000}$ codewords. Such an astronomical number of codewords makes the brute force solution of (1.14) impossible. Hence, the selection of $\mathbf{G}$ or $\mathbf{H}$ according to the Elias or Gallager ensembles, while yielding linear structure, does not by itself guarantee that a computationally efficient decoder will exist. Modern methods of coding—low-density parity-check codes, spatially coupled codes, Polar codes—while being linear codes also design additional structure into the code ensemble with the express intent to be compatible with computationally tractable decoding algorithms. To summarize, in coding

[4] We comment that this same syndrome decoding can also be used to provide a solution to the near-lossless source coding problem of Sec. 1.2. One pre-multiplies the source sequence by the parity-check matrix $\mathbf{H}$, and stores the syndrome of the source sequence. For a biased binary source, one can solve (1.14) to recover the source sequence with high probability. This approach does not feature prominently in source coding, with the exception of **distributed source coding** where it plays a prominent role. See [7, 9] for further discussion.

theory the design of a channel coding scheme involves the joint design of the codebook and the decoding algorithm.

In the point of view of the phase transitions discovered by Shannon for source and channel coding, a very interesting code construction is Erdal Arikan's Polar codes [16]. Another tractable code construction that connects to phase transitions is the spatial-coupling concept used in convolutionally structured LDPC codes [17, 18, 19]. In [16] Arikan considers symmetric channels and introduces a symmetry-breaking transformation. This transformation is a type of pre-coding that combines pairs of symmetric channels to produce a pair of virtual channels. One virtual channel is "less noisy" than the original channel and one is more noisy. Arikan then applies this transformation recursively. In the limit the virtual channels polarize. They either become noiseless and so have capacity one, or become useless and have capacity zero. Arikan shows that, in the limit, the fraction of virtual channels that become noiseless is equal to the capacity of the original symmetric channel; e.g., $1 - H_B(p)$, if the original channel were the BSC. One transmits bits uncoded over the noiseless virtual channels, and does not use the useless channels. The recursive construction yields log-linear complexity in encoding and decoding, $O(n \log n)$, making Polar codes computationally attractive. In many ways the construction is information-theoretic in nature, focusing on mutual information rather than Hamming distance as the quantity of importance in the design of capacity-achieving codes.

To conclude, we note that many overarching concepts, such as fundamental limits, achievability results, converse results, and computational limitations, that arise in classical information theory also arise in modern data science problems. In classical information theory, such notions have traditionally been considered in the context of data compression and transmission, as we have seen, whereas in data science similar notions are being studied in the realms of acquisition, data representation, analysis, and processing. There are some instances where one can directly borrow classical information-theoretic tools used to determine limits in e.g. the channel coding problem to also compute limits in data science tasks. For example, in compressive sensing [20] and group testing [21] achievability results have been derived using the probabilistic method and converse results have been developed using Fano's inequality [22]. However, there are various other data science problems where information-theoretic methods have not yet been directly applied. We elaborate further in the following sections how information-theoretic ideas, tools, and methods are also gradually shaping data science.

1.5 Connecting Information Theory to Data Science

Data science—a loosely defined concept meant to bring together various problems studied in statistics, machine learning, signal processing, harmonic analysis, and computer science under a unified umbrella—involves numerous other challenges that go beyond the traditional source coding and channel coding problems arising

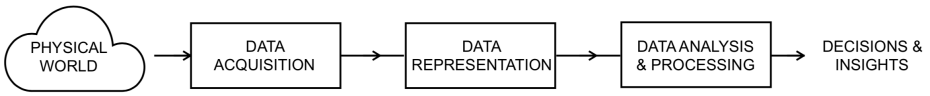

Figure 1.3 A simplified data science pipeline encompassing functions such as data acquisition, data representation, and data analysis & processing.

in communication or storage systems. These challenges are associated with the need to acquire, represent, and analyze information buried in data in a reliable and computationally efficient manner in the presence of a variety of constraints such as security, privacy, fairness, hardware resources, power, noise, and many more.

Figure 1.3 presents a typical data science pipeline, encompassing functions such as data acquisition, data representation, and data analysis, whose overarching purpose is to turn data captured from the physical-world into insights for decision making. It is also common to consider various other functions within a data science "system" such as data preparation, data exploration, and more, but we restrict ourselves to this simplified version because it serves to illustrate how information theory is helping shape data science. The goals of the different blocks of the data science pipeline in Fig. 1.3 are as follows:

- The data acquisition block is often concerned with the act of turning physical-world continuous-time analog signals into discrete-time digital signals for further digital processing.
- The data representation block concentrates on the extraction of relevant attributes from the acquired data for further analysis.
- The data analysis block concentrates on the extraction of meaningful actionable information from the data features for decision making.

Based on the description of these goals, one might think that information theory—a field that arose out of the need to study communication systems in a principled manner—has little to offer to the principles of data acquisition, representation, analysis or processing. But it turns out that information theory has been advancing our understanding of data science in three major ways:

- First, information theory has been leading to new system architectures for the different elements of the data science pipeline. Representative examples associated with new architectures for data acquisition are overviewed in Sec. 1.6.
- Second, information-theoretic methods can be used to unveil fundamental operational limits in various data science tasks, including in data acquisition, representation, analysis, and processing. Examples are overviewed in Sec. 1.6, 1.7 and 1.8.

- Third, information-theoretic measures can be used as the basis for developing algorithms for various data science tasks. We allude to some examples in Sec. 1.7 and 1.8.

In fact, the questions one can potentially ask about the data science pipeline depicted in Fig. 1.3 exhibit many parallels to the questions one asks about the communications system architecture shown in Fig. 1.1. Specifically: What are the tradeoffs in performance incurred by adopting this data science architecture? Particularly, are there other systems that do not involve the separation of the different data science elements and exhibit better performance? Are there limits in data acquisition, representation, analysis and processing? Are there computationally feasible algorithms for data acquisition, representation, analysis and processing that attain such limits?

There has been progress in data science on all three of these directions. As a concrete example that showcases many similarities between the data compression and communication problems and data science problems, information-theoretic methods have been casting insight onto the various operational regimes associated with the different data science tasks: (1) The regime where there is no algorithm—regardless of its complexity—that can perform the desired task subject to some accuracy; this "regime of impossibility" in data science has the flavor of converse results in source coding and channel coding in information theory. (2) The regime where there are algorithms, potentially very complex and computationally infeasible, that can perform the desired task subject to some accuracy; this "regime of possibility" is akin to the achievability results in source coding and channel coding. And (3) the regime where there are computationally feasible algorithms to perform the desired task subject to some accuracy; this "regime of computational feasibility" in data science has many characteristics that parallel those in design of computationally tractable source and channel coding schemes in information theory.

Interestingly, in the same way that the classical information-theoretic problems of source coding and channel coding exhibit phase transitions, many data science problems have also been shown to exhibit sharp phase transitions in the **high-dimensional setting** where the number of data samples and the number of data dimensions approach infinity. Such phase transitions are typically expressed as a function of various parameters associated with the data science problem. The resulting *information-theoretic* limit/threshold/barrier (*aka*, **statistical phase transition**) partitions the problem parameter space into two regions [23, 24, 25]: one defining problem instances that are impossible to solve and another defining problem instances that can be solved (perhaps only with a brute-force algorithm). In turn, the *computational* limit/threshold/barrier (*aka*, **computational phase transition**) partitions the problem parameter space into a region associated with problem instances that are easy to solve and another associated with instances that are hard to solve [26, 27].

There can however be differences in how one establishes converse and achiev-

ability results—and thereon phase transitions—in classical information-theoretic problems and data science ones. Converse results in data science can often be established using Fano's inequality or variations (see also Chapter 16). In contrast, achievability results often cannot rely on classical techniques, such as the probabilistic method, necessitating instead the direct analysis of the algorithms. Chapter 13 elaborates on some emerging tools that may be used to establish statistical and computational limits in data science problems.

In summary, numerous information-theoretic tools, methods, and quantities are increasingly becoming essential to cast insight onto data science. It is impossible to capture all the recent developments in a single chapter, but the following sections sample a number of recent results under three broad themes: data acquisition, data representation, and data analysis and processing.

1.6 Information Theory and Data Acquisition

Data acquisition is a critical element of the data science architecture shown in Fig. 1.3. It often involves the conversion of a **continuous-time analog** signal into a **discrete-time digital** signal that can be further processed in digital signal processing pipelines.[5]

Conversion of a continuous-time analog signal $x(t)$ into a discrete-time digital representation typically entails two operations. The first operation—known as sampling—involves recording the values of the original signal $x(t)$ at particular instants of time. The simplest form of sampling is direct uniform sampling in which the signal is recorded at uniform sampling times $x(kT_s) = x(k/F_s)$ where T_s denotes the **sampling period** (in seconds), F_s denotes the **sampling frequency** (in Hertz), and k is an integer. Another popular form of sampling is generalized shift-invariant sampling in which $x(t)$ is first filtered by a linear time-invariant (LTI) filter, or a bank of LTI filters, and only then sampled uniformly [28]. Other forms of generalized and nonuniform sampling have also been studied. Surprisingly, under certain conditions, the sampling process can be shown to be lossless: for example, the classical **sampling theorem** for bandlimited processes asserts that it is possible to perfectly recover the original signal from its uniform samples provided that the sampling frequency F_s is at least twice the signal bandwidth B. This minimal sampling frequency $F_{\mathrm{NQ}} = 2B$ is referred to as the **Nyquist rate** [28].

The second operation—known as **quantization**—involves mapping the continuous-valued signal samples onto discrete-valued ones. The levels are taken from a finite set of levels that can be represented using a finite sequence of bits. In (optimal) **vector quantization** approaches, a series of signal samples are converted simultaneously to a bit sequence, whereas in (sub-optimal) **scalar quantization**, each individual sample is mapped to bits. The quantization process is

[5] Note that it is also possible that the data is already presented in an inherently digital format; Chapters 3, 4, and 6 deal with such scenarios.

inherently lossy since it is impossible to accurately represent real-valued samples using a finite set of bits. **Rate-distortion theory** establishes a tradeoff between the average number of bits used to encode each signal sample—referred to as the **rate**—and the average distortion incurred in the reconstruction of each signal sample—referred to simply as the **distortion**—via two functions. The **rate-distortion function** $R(D)$ specifies the smallest number of bits required on average per sample when one wishes to represent each sample with average distortion less than D, whereas the **distortion-rate function** $D(R)$ specifies the lowest average distortion achieved per sample when one wishes to represent on average each sample with R bits [7]. A popular measure of distortion in the recovery of the original signal samples from the quantized ones is the mean-squared error (MSE). Note that this class of problems—known as lossy source coding—is the counterpart of the lossless source coding problems discussed earlier.

The motivation for this widely-used data acquisition architecture, involving (1) a sampling operation at or just above the Nyquist rate and (2) scalar or vector quantization operations, is its simplicity that leads to a practical implementation. However, it is well known that the separation of the sampling and quantization operations is not necessarily optimal. Indeed, the optimal strategy that attains Shannon's distortion-rate function associated with arbitrary continuous-time random signals with known statistics involves a general mapping from continuous-time signal space to a sequence of bits that does not consider any practical constraints in its implementation [1, 3, 29]. Therefore, recent years have witnessed various generalizations of this data acquisition paradigm informed by the principles of information theory, on the one hand, and guided by practical implementations, on the other.

One recent extension considers a data acquisition paradigm that illuminates the dependency between these two operations [30, 31, 32]. In particular, given a total rate budget, Kipnis et al. [30, 31, 32] draw on information-theoretic methods to study the lowest sampling rate required to sample a signal such that the reconstruction of the signal from the bit-constrained representation of its samples results in minimal distortion. The sampling operation consists of an LTI filter, or bank of filters, followed by pointwise sampling of the outputs of the filters. The authors also show that, without assuming any particular structure on the input analog signal, this sampling rate is often below the signal's Nyquist rate. That is, due to the fact that there is loss encountered by the quantization operation, there is no longer in general a requirement to sample the signal at the Nyquist rate.

As an example, consider the case where $x(t)$ is a stationary random process bandlimited to B with a triangular power spectral density (PSD) given formally by

$$S(f) = \frac{\sigma_x^2}{B}[1 - |f/B|]_+ \tag{1.15}$$

with $[a]_+ = \max(a, 0)$. In this case, the Nyquist sampling rate is $2B$. However,

when quantization is taken into account, the sampling rate can be lowered without introducing further distortion. Specifically, assuming a bit rate leading to distortion D, the minimal sampling rate is shown to be equal to [32]:

$$f_R = 2B\sqrt{1 - D/\sigma_x^2}. \tag{1.16}$$

Thus, as the distortion grows, the minimal sampling rate is reduced. When we do not allow any distortion, namely, no quantization takes place, $D = 0$ and $f_R = 2B$ so that Nyquist rate sampling is required.

Such results show how information-theoretic methods are leading to new insights about the interplay between sampling and quantization. In particular, these new results can be seen as an extension of the classical sampling theorem applicable to bandlimited random processes in the sense that they describe the minimal amount of excess distortion in the reconstruction of a signal due to lossy compression of its samples, leading to the minimal sampling frequency required to achieve this distortion.[6] In general, this sampling frequency is below the Nyquist rate. Chapter 2 surveys some of these recent results in data acquisition.

Another generalization of the classical data acquisition paradigm considers scenarios where the end goal is not to reconstruct the original analog signal $x(t)$ but rather perform some other operation on it [33]. For example, in the context of parameter estimation, Rodrigues et al. [33] show that the number of bits per sample required to achieve a certain distortion in such **task-oriented** data acquisition can be much lower than that required for **task-ignorant** data acquisition. More recently, Shlezinger et al. [34, 35] study task-oriented **hardware-efficient** data acquisition systems, where optimal vector quantizers are replaced by practical ones. Even though the optimal rate-distortion curve cannot be achieved by replacing optimal vector quantizers by simple serial scalar ones, it is shown in [34, 35] that one can get close to the minimal distortion in settings where the information of interest is not the signal itself, but rather a low dimensional parameter vector embedded in the signal. A practical application of this setting is to massive multiple-input multiple-output (MIMO) systems where there is a strong need to utilize simple low-resolution quantizers due to power and memory constraints. In this context, it is possible to design a simple quantization system, consisting of scalar uniform quantizers and linear pre- and post-processing, leading to minimal channel estimation distortion.

These recent results also showcase how information-theoretic methods can cast insight into the interplay between data acquisition, representation, and analysis, in the sense that knowledge of the data analysis goal can influence the data acquisition process. These results therefore also suggest new architectures for the conventional data science pipeline that do not involve a strict separation between the data acquisition, data representation, and data analysis & processing blocks.

Beyond this data acquisition paradigm involving the conversion of continuous-

[6] In fact, this theory can be used even when the input signal is not bandlimited.

time signals to digital ones, recent years have also witnessed the emergence of various other data acquisition approaches. Chapters 3, 4, and 6 cover further data acquisition strategies that are also benefiting from information-theoretic methods.

1.7 Information Theory and Data Representation

The outputs of the data acquisition block—often known as **"raw" data**—need to be typically turned into "meaningful" representations—known as **features**—for further data analysis. Note that the act of transforming raw data into features, where the number of dimensions in the features is lower than that in the raw data, is also referred to as **dimensionality reduction**.

Recent years have witnessed a shift from **model-based data representations**, relying on pre-determined transforms—such as wavelets, curvelets, and shearlets—to compute the features from raw data, to **data-driven representations** that leverage a number of (raw) data "examples"/"instances" to first learn a (linear or nonlinear) data representation transform conforming to some postulated data generative model [36, 37, 38, 39]. Mathematically, given N (raw) data examples $\{\mathbf{y}^i \in \mathcal{Y}\}_{i=1}^N$ (referred to as **training samples**), data-driven representation learning often assumes generative models of the form

$$\mathbf{y}^i = F(\mathbf{x}^i) + \mathbf{w}^i, \quad i = 1, \ldots, N, \tag{1.17}$$

where $\mathbf{x}^i \in \mathcal{X}$ denote feature vectors that are assumed to be realizations of a random vector $\mathbf{X}$ distributed as p_X, $\mathbf{w}^i$ denote acquisition noise and/or modeling errors that are realizations of random noise $\mathbf{W}$ distributed as p_W, and $F : \mathcal{X} \to \mathcal{Y}$ denotes the true (linear or nonlinear) representation transform that belongs to some postulated class of transforms $\mathcal{F}$. The operational challenge in representation learning is to estimate $F(\cdot)$ using the training samples, after which the features can be obtained using the inverse images of data samples returned by

$$G \stackrel{\text{def}}{=} F^{-1} : \mathcal{Y} \to \mathcal{X}. \tag{1.18}$$

Note that if $F(\cdot)$ is not a bijection then $G(\cdot)$ will not be the inverse operator.[7]

In the literature, $F(\cdot)$ and $G(\cdot)$ are sometimes also referred to as the **synthesis operator** and the **analysis operator**, respectively. In addition, the representation learning problem as stated is referred to as **unsupervised representation learning** [37, 39]. Another category of representation learning is **supervised representation learning** [40, 41], in which training data correspond to tuples $(\mathbf{y}^i, \ell^i)$ with ℓ^i termed the **label** associated with training sample $\mathbf{y}^i$. Representation learning in this case involves obtaining an analysis/synthesis operator that results in the best task-specific (e.g., classification and regression) performance.

[7] In some representation learning problems, instead of using $F(\cdot)$ to obtain the inverse images of data samples, $G(\cdot)$ is learned directly from training samples.

Another major categorization of representation learning is in terms of the linearity of $G(\cdot)$, with the resulting classes referred to as **linear representation learning** and **nonlinear representation learning**, respectively.

The problem of learning (estimating) the true transformation from a given postulated generative model poses various challenges. One relates to the design of appropriate algorithms for estimating $F(\cdot)$ and computing inverse images $G(\cdot)$. Another challenge involves understanding information-theoretic and computational limitations in representation learning in order to identify regimes where existing algorithms are nearly optimal, regimes where existing algorithms are clearly sub-optimal, and to guide the development of new algorithms. These challenges are also being addressed using information-theoretic tools. For example, researchers often map the representation learning problem onto a channel coding problem, where the transformation $F(\cdot)$ represents the message that needs to be decoded at the output of a channel that maps $F(\cdot)$ onto $F(\mathbf{X}) + \mathbf{W}$. This allows leveraging information-theoretic tools such as Fano's Inequality for derivation of fundamental limits on the estimation error of $F(\cdot)$ as a function of the number of training samples [42, 43, 44, 45, 25]. We next provide a small sampling of representation learning results that involve the use of information-theoretic tools and methods.

1.7.1 Linear representation learning

Linear representation learning constitutes one of the oldest and, to this date, the most prevalent data representation technique in data science. While there are several different variants of linear representation learning in both the unsupervised and the supervised settings, all these variants are based on the assumption that the raw data samples lie near a **low-dimensional (affine) subspace**. Representation learning in this case is therefore equivalent to learning the subspace(s) underlying raw data. This will be a single subspace in the unsupervised setting, as in **Principal Component Analysis (PCA)** [46, 47, 48] and **Independent Component Analysis (ICA)** [49, 50], and multiple subspaces in the supervised setting, as in **Linear Discriminant Analysis (LDA)** [51, 52, 53] and **Quadratic Discriminant Analysis (QDA)** [53].

Mathematically, linear representation learning operates under the assumption of the raw data space being $\mathcal{Y} = \mathbb{R}^m$, the feature space being $\mathcal{X} = \mathbb{R}^k$ with $k \ll m$, the raw data samples given by

$$\mathbf{Y} = \mathbf{A}\mathbf{X} + \mathbf{W} \tag{1.19}$$

with $\mathbf{A} \in \mathcal{F} \subset \mathbb{R}^{m \times k}$, and the feature estimates being given by $\widehat{\mathbf{X}} = \mathbf{B}\mathbf{Y}$ with $\mathbf{B} \in \mathbb{R}^{k \times m}$. In this setting, $(F, G) = (\mathbf{A}, \mathbf{B})$ and representation learning reduces to estimating the linear operators $\mathbf{A}$ and/or $\mathbf{B}$ under various assumptions on $\mathcal{F}$ and the generative model.[8] In the case of PCA, for example, it is assumed that

[8] Supervised learning typically involves estimation of multiple $\mathbf{A}$'s and/or $\mathbf{B}$'s.

$\mathcal{F}$ is the Stiefel manifold in $\mathbb{R}^m$ and the feature vector $\mathbf{X}$ is a random vector that has zero mean and uncorrelated entries. On the other hand, ICA assumes $\mathbf{X}$ to have zero mean and independent entries. (The zero-mean assumption in both PCA and ICA is for ease of analysis and can be easily removed at the expense of extra notation.)

Information-theoretic frameworks have long been used to develop computational approaches for estimating $(\mathbf{A}, \mathbf{B})$ in ICA and its variants; see, e.g., [49, 54, 50, 55]. Recent years have also seen the use of information-theoretic tools such as Fano's Inequality to derive sharp bounds on the feasibility of linear representation learning. One such result that pertains to PCA under the so-called **spiked covariance model** is described next.

Suppose the training data samples are N independent realizations according to (1.19), i.e.,

$$\mathbf{y}^i = \mathbf{A}\mathbf{x}^i + \mathbf{w}^i, \quad i = 1, \ldots, N, \tag{1.20}$$

where $\mathbf{A}^T\mathbf{A} = \mathbf{I}$ and both $\mathbf{x}^i$ and $\mathbf{w}^i$ are independent realizations of $\mathbf{X}$ and $\mathbf{W}$ that have zero mean and diagonal covariance matrices given by

$$\mathbb{E}[\mathbf{X}\mathbf{X}^T] = \mathrm{diag}(\lambda_1, \ldots, \lambda_k), \quad \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_k > 0, \tag{1.21}$$

and $\mathbb{E}[\mathbf{W}\mathbf{W}^T] = \sigma^2\mathbf{I}$, respectively. Note that the ideal $\mathbf{B}$ in this PCA example is given by $\mathbf{B} = \mathbf{A}^T$. It is then shown in [43, Theorem 5] using various analytical tools, which include Fano's Inequality, that $\mathbf{A}$ can be reliably estimated from N training samples only if[9]

$$\frac{N\lambda_k^2}{k(m-k)(1+\lambda_k)} \to \infty. \tag{1.22}$$

This is the "converse" for the spiked covariance estimation problem.

The "achievability" result for this problem is also provided in [43]. Specifically, when the condition given in (1.22) is satisfied, a practical algorithm exists that allows for reliable estimation of $\mathbf{A}$ [43]. This algorithm involves taking $\widehat{\mathbf{A}}$ to be the k eigenvectors corresponding to the k largest eigenvalues of the sample covariance $\frac{1}{N}\sum_{i=1}^N \mathbf{y}^i\mathbf{y}^{iT}$ of the training data. We therefore have a sharp information-theoretic phase transition in this problem, which is characterized by (1.22). Notice here, however, that while the converse makes use of information-theoretic tools, the achievability result does not involve the use of the probabilistic method; rather, it requires analysis of an explicit (deterministic) algorithm.

The sharp transition highlighted by the aforementioned result can be interpreted in various ways. One of the implications of this result is that it is impossible to reliably estimate the PCA features when $m > N$ and $m, N \to \infty$. In such **high-dimensional PCA** settings, it is now well understood that **sparse PCA**, in which the columns of $\mathbf{A}$ are approximately "sparse," is more appropriate for

[9] Reliable estimation here means that the error between $\widehat{\mathbf{A}}$ and $\mathbf{A}$ converges to 0 with increasing number of samples.

linear representation learning. We refer the reader to works such as [43, 56, 57] that provide various information-theoretic limits for the sparse PCA problem.

We conclude by noting that there has been some recent progress on bounds on the computational feasibility of linear representation learning. For example, the fact that there is a practical algorithm to learn a linear data representation in some high-dimensional settings implies that computational barriers can almost coincide with information-theoretic ones. It is important to emphasize though that recent work—applicable to the detection of a subspace structure within a data matrix [58, 59, 60, 61, 25, 62]—has revealed that classical computationally feasible algorithms such as PCA cannot always approach the information-theoretic detection threshold [61, 25].

1.7.2 Nonlinear representation learning

While linear representation learning techniques tend to have low computational complexity, they often fail to capture relevant information within complex physical phenomena. This, coupled with meteoric rise in computing power, has led to widespread adoption of nonlinear representation learning in data science.

There is a very wide portfolio of nonlinear representation techniques, but one of the most well-known classes, which has been the subject of much research during the last two decades, postulates that (raw) data lie near a **low-dimensional manifold** embedded in a higher-dimensional space. Representation learning techniques belonging to this class include local linear embedding [63], Isomap [64], kernel entropy component analysis (ECA) [65], and nonlinear generalizations of linear techniques using the **kernel trick** (e.g., kernel PCA [66], kernel ICA [67], and kernel LDA [68]). The use of information-theoretic machinery in these methods has mostly been limited to formulations of the algorithmic problems, as in kernel ECA and kernel ICA. While there exist some results that characterize the regime in which manifold learning is impossible, such results leverage the probabilistic method rather than more fundamental information-theoretic tools [69].

Recent years have seen the data science community widely embrace another nonlinear representation learning approach that assumes data lie near a **union of subspaces (UoS)**. This approach tends to have several advantages over manifold learning because of the linearity of individual components (subspaces) in the representation learning model. While there exist methods that learn the subspaces explicitly, one of the most popular classes of representation learning under the UoS model in which the subspaces are implicitly learned is referred to as **dictionary learning** [38]. Formally, dictionary learning assumes the data space to be $\mathcal{Y} = \mathbb{R}^m$, the feature space to be

$$\mathcal{X} = \{\mathbf{x} \in \mathbb{R}^p : \|\mathbf{x}\|_0 \leq k\} \tag{1.23}$$

with $k \ll m \leq p$, and the random generative model to be

$$\mathbf{Y} = \mathbf{D}\mathbf{X} + \mathbf{W} \tag{1.24}$$

with $\mathbf{D} \in \mathcal{F} = \{\mathbf{D} \in \mathbb{R}^{m \times p} : \|\mathbf{D}_i\|_2 = 1, i = 1, \ldots, p\}$ representing a dictionary and $\mathbf{W} \in \mathbb{R}^m$ representing the random noise vector. This corresponds to the random data vector $\mathbf{Y}$ lying near a union of $\binom{p}{k}$ k-dimensional subspaces. Notice that while $F(\cdot) = \mathbf{D}$ is a linear operator,[10] its inverse image $G(\cdot)$ is highly nonlinear and typically computed as

$$\widehat{\mathbf{X}} = G(\mathbf{Y}) = \arg \min_{\mathbf{X}: \|\mathbf{X}\|_0 \leq k} \|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_2^2, \tag{1.25}$$

with (1.25) referred to as **sparse coding**.

The last fifteen years have seen the development of a number of algorithms that enable learning of the dictionary $\mathbf{D}$ in both unsupervised and supervised settings [70, 71, 41, 72]. Sample complexity of these algorithm in terms of both infeasibility (converse) and achievability has been a more recent effort [73, 74, 75, 76, 44, 45]. In particular, it is established in [44] using Fano's Inequality that the number of samples N, which are independent realizations of the generative model (1.24), i.e.,

$$\mathbf{y}^i = \mathbf{D}\mathbf{x}^i + \mathbf{w}^i, \quad i = 1, \ldots, N, \tag{1.26}$$

must scale at least as fast as $N = O(mp^2\epsilon^{-2})$ in order to ensure recovery of an estimate $\widehat{\mathbf{D}}$ such that $\|\widehat{\mathbf{D}} - \mathbf{D}\|_F \leq \epsilon$. This lower bound on sample complexity, which is derived in the minimax sense, is akin to the converse bounds in source and channel coding in classical information theory. However, general tightness of this lower bound, which requires analyzing explicit (deterministic) dictionary learning algorithms and deriving matching achievability results, remains an open problem. Computational limits are also in general open for dictionary learning.

Recent years have also seen extension of these results to the case of data that has a multidimensional (tensor) structure [45]. We refer the reader to Chapter 5 in the book for a more comprehensive review of dictionary learning results pertaining to both vector and tensor data.

Linear representation learning, manifold learning, and dictionary learning are all based on a geometric viewpoint of data. It is also possible to view these representation learning techniques from a purely numerical linear algebra perspective. Data representations in this case are referred to as **matrix factorization-based representations**. The matrix factorization perspective of representation learning allows one to expand the classes of learning techniques by borrowing from the rich literature on linear algebra. Non-negative matrix factorization [77], for instance, allows one to represent data that are inherently non-negative in terms of non-negative features that can be assigned physical meanings. We refer the reader to [78] for a more comprehensive overview of matrix factorizations in data

[10] Strictly speaking, $\mathbf{D}$ restricted to $\mathcal{X}$ is also nonlinear.

science; [79] also provides a recent information-theoretic analysis of nonnegative matrix factorization.

1.7.3 Recent trends in representation learning

Beyond the subspace and UoS models described above, another emerging approach to learning data representations relates to the use of **deep neural networks** [80]. In particular, this involves designing a nonlinear transformation $G : \mathcal{Y} \to \mathcal{X}$ consisting of a series of stages, with each stage encompassing a linear and a nonlinear operation, that can be used to produce a data representation $\mathbf{x} \in \mathcal{X}$ given a data instance $\mathbf{y} \in \mathcal{Y}$ as follows:

$$\mathbf{x} = G(\mathbf{y}) = f_L(\mathbf{W}^L \cdot f_{L-1}(\mathbf{W}^{L-1} \cdot (\cdots f_1(\mathbf{W}^1\mathbf{y} + \mathbf{b}^1) \cdots) + \mathbf{b}^{L-1}) + \mathbf{b}^L) \quad (1.27)$$

where $\mathbf{W}^i \in \mathbb{R}^{n_i \times n_{i-1}}$ is a weight matrix, $\mathbf{b}^i \in \mathbb{R}^{n_i}$ is a bias vector, $f_i : \mathbb{R}^{n_i} \to \mathbb{R}^{n_i}$ is a nonlinear operator such as a **rectified linear unit (ReLU)**, and L corresponds to the number of layers in the deep neural network. The challenge then relates to how to learn the set of weight matrices and bias vectors associated with the deep neural network. For example, in classification problems where each data instance $\mathbf{x}$ is associated with a discrete label ℓ, one typically relies on a training set $(\mathbf{y}^i, \ell^i), i = 1, \ldots, N$, to define a loss function that can be used to tune the various parameters of the network using algorithms such as gradient descent or stochastic gradient descent [81].

This approach to data representation underlies some of the most spectacular advances in areas such as computer vision, speech recognition, speech translation, natural language processing, and many more, but this approach is also not fully understood. However, information-theoretic oriented studies have also been recently conducted to cast insight onto the performance of deep neural networks by enabling the analysis of the learning process or the design of new learning algorithms. For example, Tishby et al. [82] propose an information-theoretic analysis of deep neural networks based on the **information bottleneck** principle. They view the neural network learning process as a tradeoff between compression and prediction that leads up to the extraction of a set of minimal sufficient statistics from the data in relation to the target task. Shwartz-Ziv et al. [83]—building upon the work in [82]—also propose an information bottleneck based analysis of deep neural networks. In particular, they study **information paths** in the so-called **information plane** capturing the evolution of a pair of mutual informations over the network during the training process: one relates to the mutual information between the i-th layer output and the target data label, and the other corresponds to the mutual information between the i-th layer output and the data itself. They also demonstrate empirically that the widely used stochastic gradient descent algorithm undergoes a "fitting" phase—where the mutual information between the data representations and the target data label increases—and a "compression" phase—where the mutual information be-

tween the data representations and the data decreases. See also related works investigating the flow of information in deep networks [84, 85, 86, 87].

Achille and Soatto [88] also use an information-theoretic approach to understand deep neural networks based data representations. In particular, they show how deep neural networks can lead to minimal sufficient representations with properties such as invariance to nuisances, and provide bounds that connect the amount of information in the weights and the amount of information in the activations to certain properties of the activations such as invariance. They also show that a new information bottleneck Lagrangian involving the information between the weights of a network and the training data can overcome various overfitting issues.

More recently, information-theoretic metrics have been used as a proxy to learn data representations. In particular, Hjelm et al. [89] propose unsupervised learning of representations by maximizing the mutual information between an input and the output of a deep neural network.

In summary, this body of work suggests that information-theoretic quantities such as mutual information can inform the analysis, design, and optimization of state-of-the-art representation learning approaches. Chapter 11 covers some of these recent trends in representation learning.

1.8 Information Theory and Data Analysis & Processing

The outputs of the data representation block—the features—are often the basis for further data analysis or processing, encompassing both **statistical inference** and **statistical learning** tasks such as estimation, regression, classification, clustering, and many more.

Statistical inference forms the core of classical statistical signal processing and statistics. Broadly speaking, it involves use of explicit stochastic data models to understand various aspects of data samples (features). These models can be **parametric**, defined as those characterized by a finite number of parameters, or **nonparametric**, in which the number of parameters continuously increases with the number of data samples. There is a large portfolio of statistical inference tasks, but we limit our discussion to the problems of **model selection**, **hypothesis testing**, **estimation**, and **regression**.

Briefly, model selection involves the use of data features/samples to select a stochastic data model from a set of candidate models. Hypothesis testing, on the other hand, is the task of determining whether a certain postulated hypothesis (stochastic model) underlying the data is true or false. This is referred to as *binary* hypothesis testing, as opposed to *multiple* hypothesis testing in which the data are tested against several hypotheses. Statistical estimation, often studied under the umbrella of **inverse problems** in many disciplines, is the task of inferring some parameters underlying the stochastic data model. In contrast, regression involves estimating the relationships between different data features

that are divided into the categories of **response variable(s)** (also known as **dependent variables**) and **predictors** (also known as **independent variables**).

Statistical learning, along with **machine learning**, primarily concentrates on approaches to find structure in data. In particular, while the boundary between statistical inference and statistical learning is not a hard one, statistical learning tends not to focus on explicit stochastic models of data generation; rather, it often treats the data generation mechanism as a black box, and primarily concentrates on learning a "model" with good prediction accuracy [90]. There are two major paradigms in statistical learning: **supervised learning** and **unsupervised learning**.

In supervised learning, one wishes to determine predictive relationships between and/or across data features. Representative supervised learning approaches include classification problems where the data features are mapped to a discrete set of values (a.k.a. **labels**) and regression problems where the data features are mapped instead to a continuous set of values. Supervised learning often involves two distinct phases of **training** and **testing**. The training phase involves use of a dataset, referred to as **training data**, to learn a model that finds the desired predictive relationship(s). These predictive relationships are often implicitly known in the case of training data and the goal is to leverage this knowledge for learning a model during training that *generalizes* these relationships to as-yet unseen data. Often, one also employs a **validation dataset** in concert with training data to tune possible **hyper-parameters** associated with a statistical learning model. The testing phase involves use of another dataset with known characteristics, termed **test data**, to estimate the learned model's generalization capabilities. The error incurred by the learned model on training and test data is referred to as **training error** and **testing error**, respectively, while the error that the model will incur on future unseen data can be captured by the so-called **generalization error**. One of the biggest challenges in supervised learning is understanding the generalization error of a statistical learning model as a function of the number of data samples in training data.

In unsupervised learning, one wishes instead to determine the underlying structure within the data. Representative unsupervised learning approaches include **density estimation**, where the objective is to determine the underlying data distribution given a set of data samples, and **clustering**, where the aim is to organize the data points onto different groups so that points belonging to the same group exhibit some degree of similarity and points belonging to different groups are distinct.

Challenges arising in statistical inference and learning also involve analyzing, designing and optimizing inference and learning algorithms, and understanding statistical and computational limits in inference and learning tasks. We next provide a small sampling of statistical inference and statistical learning results that involve the use of information-theoretic tools and methods.

1.8.1 Statistical inference

We now survey some representative results arising in model selection, estimation, regression, and hypothesis testing problems that benefit from information-theoretic methods. We also offer an example associated with community detection and recovery on graphs where information-theoretic and related tools can be used to determine statistical and computational limits.

Model selection

On the algorithmic front, the problem of model selection has been largely impacted by information-theoretic tools. *Given a data set, which statistical model "best" describes the data?* A huge array of work, dating back to the 70's, has tackled this question using various information-theoretic principles. The **Akaike Information Criterion (AIC)** for model selection [91], for instance, uses the KL divergence as the main tool for derivation of the final criterion. The **Minimum Description Length (MDL)** principle for model selection [92], on the other hand, makes a connection between source coding and model selection and seeks a model that best compresses the data. AIC and MDL principles are just two of a number of information-theoretic inspired model selection approaches; we refer the interested reader to Chapter 12 for further discussion.

Estimation and regression

Over the years, various information-theoretic tools have significantly advanced our understanding of the interconnected problems of estimation and regression. In statistical inference, a typical estimation/regression problem involving a scalar random variable $Y \in \mathbb{R}$ takes the form

$$Y = f(\mathbf{X}; \boldsymbol{\beta}) + W, \tag{1.28}$$

where the random vector $\mathbf{X} \in \mathbb{R}^p$ is referred to as a **covariate** in statistics and **measurement vector** in signal processing, $\boldsymbol{\beta} \in \mathbb{R}^p$ denotes the unknown parameter vector, termed **regression parameters** and **signal** in statistics and signal processing, respectively, and W represents observation noise/modeling error.[11] Both estimation and regression problems in statistical inference concern themselves with recovering $\boldsymbol{\beta}$ from N realizations $\{(y^i, \mathbf{x}^i)\}_{i=1}^N$ from the model (1.28) under an assumed $f(\cdot;\cdot)$. In estimation, one is interested in recovering a $\widehat{\boldsymbol{\beta}}$ that is as close to the true $\boldsymbol{\beta}$ as possible; in regression, on the other hand, one is concerned with prediction, i.e., how close $f(\mathbf{X}; \widehat{\boldsymbol{\beta}})$ is to $f(\mathbf{X}; \boldsymbol{\beta})$ for the random vector $\mathbf{X}$. Many modern setups in estimation/regression problems correspond to the high-dimensional setting in which $N \ll p$. Such setups often lead to seemingly ill-posed mathematical problems, resulting in the following important question: *How small can the estimation and/or regression errors be as a function of N, p and properties of the covariates and parameters.*

[11] The assumption is that raw data has been transformed into its features, which correspond to $\mathbf{X}$.

Information-theoretic methods have been used in a variety of ways to address this question for a number of estimation/regression problems. The most well known of these results are for the **Generalized Linear Model (GLM)** where the realizations of $(Y, \mathbf{X}, \mathbf{W})$ are given by:

$$y^i = \mathbf{x}^{i^T}\boldsymbol{\beta} + w^i \Longrightarrow \mathbf{y} = \widetilde{\mathbf{X}}\boldsymbol{\beta} + \mathbf{w} \tag{1.29}$$

with $\mathbf{y} \in \mathbb{R}^N, \widetilde{\mathbf{X}} \in \mathbb{R}^{N\times p}$, and $\mathbf{w} \in \mathbb{R}^N$ denoting concatenations of y^i, $\mathbf{x}^{i^T}$, and w^i, respectively. Fano's Inequality has been used to derive lower bounds on the errors in GLMs under various assumptions on the matrix $\widetilde{\mathbf{X}}$ and $\boldsymbol{\beta}$ [93, 94, 95]. Much of this work has been limited to the case of **sparse** $\boldsymbol{\beta}$, in which it is assumed that no more than a few (say, $s \ll N$) regression parameters are nonzero [93, 94]. The work by Raskutti et al. [95] extends many of these results to $\boldsymbol{\beta}$ that is not strictly sparse. This work focuses on *approximately* sparse regression parameters, defined as lying within an ℓ_q ball, $q \in [0, 1]$ of radius R_q as follows:

$$\mathcal{B}_q(R_q) = \{\boldsymbol{\beta} : \sum_{i=1}^{p} |\beta_i|^q \leq R_q\}, \tag{1.30}$$

and provides matching minimax lower and upper bounds (i.e., optimal minimax rate) for both estimation error, $\|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}\|_2^2$, and prediction error, $\frac{1}{n}\|\widetilde{\mathbf{X}}(\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta})\|_2^2$. In particular, it is established that, under suitable assumptions on $\widetilde{\mathbf{X}}$, it is possible to achieve estimation and prediction errors in GLMs that scale as $R_q \left(\log p/N\right)^{1-q/2}$. The corresponding result for exact sparsity can be derived by setting $q = 0$ and $R_q = s$. Further, there exist no algorithms, regardless of their computational complexity, that can achieve errors smaller than this rate for every $\boldsymbol{\beta}$ in an ℓ_q ball. As one might expect, Fano's Inequality is the central tool used by Raskutti et al. [95] to derive this lower bound (the "converse"). The achievability result requires direct analysis of algorithms, as opposed to use of the probabilistic method in classical information theory. Since both the converse and the achievability bounds coincide in regression and estimation under the GLM, we end up with a sharp statistical phase transition. Chapters 6, 7, 8, and 16 elaborate further on various other recovery and estimation problems arising in data science, along with key tools that can be used to cast insight into such problems.

Additional information-theoretic results are known for the **Standard Linear Model**—where $\mathbf{Y} = \sqrt{s}\mathbf{X}\boldsymbol{\beta} + \mathbf{W}$ with $\mathbf{Y} \in \mathbb{R}^n$, $\mathbf{X} \in \mathbb{R}^{n\times p}$, $\boldsymbol{\beta} \in \mathbb{R}^p$, $\mathbf{W} \in \mathbb{R}^n \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, and s a scaling factor representing a signal-to-noise ratio. In particular, subject to mild conditions on the distribution of the parameter vector, it has been established that the mutual information and the minimum mean-squared error obey the so-called **I-MMSE** relationship given by [96]:

$$\frac{dI\left(\boldsymbol{\beta}; \sqrt{s}\mathbf{X}\boldsymbol{\beta} + \mathbf{W}\right)}{ds} = \frac{1}{2} \cdot \mathsf{mmse}\left(\mathbf{X}\boldsymbol{\beta} | \sqrt{s}\mathbf{X}\boldsymbol{\beta} + \mathbf{W}\right) \tag{1.31}$$

where $I\left(\boldsymbol{\beta}; \sqrt{s}\mathbf{X}\boldsymbol{\beta} + \mathbf{W}\right)$ corresponds to the mutual information between the

standard linear model input and output and

$$\mathsf{mmse}\left(\mathbf{X}\boldsymbol{\beta}|\sqrt{s}\mathbf{X}\boldsymbol{\beta}+\mathbf{W}\right)=\mathbb{E}\left\{\left\|\mathbf{X}\boldsymbol{\beta}-\mathbb{E}\left\{\mathbf{X}\boldsymbol{\beta}|\sqrt{s}\mathbf{X}\boldsymbol{\beta}+\mathbf{W}\right\}\right\|_2^2\right\} \tag{1.32}$$

is the minimum mean-squared error associated with the estimation of $\mathbf{X}\boldsymbol{\beta}$ given $\sqrt{s}\mathbf{X}\boldsymbol{\beta}+\mathbf{W}$. Other relations involving information-theoretic quantities, such as mutual information, and estimation-theoretic ones have also been established in a wide variety of settings in recent years, such as Poisson models [97]. These relations have been shown to have important implications in classical information-theoretic problems – notably in the analysis and design of communications systems (e.g. [98, 99, 100, 101]) – and, more recently, in data science ones. In particular, Chapter 7 elaborates further on how the I-MMSE relationship can be used to cast insight into modern high-dimensional inference problems.

Hypothesis testing

Information-theoretic tools have also been advancing our understanding of hypothesis testing problems (one of the most widely used statistical inference techniques). In general, we can distinguish between **binary hypothesis testing** problems, where the data is tested against two hypotheses often known as the **null** and the **alternate** hypotheses, and **multiple hypothesis testing** problems in which the data is tested against multiple hypotheses. We can also distinguish between **Bayesian** approaches to hypothesis testing, where one specifies a prior probability associated with each of the hypothesis, and **non-Bayesian** ones, in which one does not specify *a priori* any prior probability.

Formally, a classical formulation of the binary hypothesis testing problem involves testing whether a number of i.i.d. data samples (features) $\mathbf{x}^1,\mathbf{x}^2,\ldots,\mathbf{x}^N$ of a random variable $\mathbf{X}\in\mathcal{X}\sim p_X$ conform to one of the following hypotheses $\mathcal{H}_0: p_X=p_0$ or $\mathcal{H}_1: p_X=p_1$, where under the first hypothesis one postulates that the data is generated i.i.d. according to model (distribution) p_0 and under the second hypothesis one assumes the data is generated i.i.d. according to model (distribution) p_1. A binary hypothesis test $T:\mathcal{X}\times\cdots\times\mathcal{X}\to\{\mathcal{H}_0,\mathcal{H}_1\}$ is a mapping that outputs an estimate of the hypothesis given the data samples.

In non-Bayesian settings, the performance of such a binary hypothesis test can be described by two **error probabilities**. The **type-I error probability**, which relates to the rejection of a true null hypothesis, is given by:

$$P_{e|0}\left(T\right)=\mathbb{P}\left(T\left(\mathbf{X}^1,\mathbf{X}^2,\ldots,\mathbf{X}^N\right)=\mathcal{H}_1|\mathcal{H}_0\right) \tag{1.33}$$

and the **type-II error probability**, which relates to the failure to reject a false null hypothesis, is given by:

$$P_{e|1}\left(T\right)=\mathbb{P}\left(T\left(\mathbf{X}^1,\mathbf{X}^2,\ldots,\mathbf{X}^N\right)=\mathcal{H}_0|\mathcal{H}_1\right). \tag{1.34}$$

In this class of problems, one is typically interested in minimizing one of the error probabilities subject to a constraint on the other error probability as follows:

$$P_e(\alpha)=\min_{T:P_{e|0}(T)\leq\alpha}P_{e|1}\left(T\right) \tag{1.35}$$

where the minimum can be achieved using the well-known **Neymann–Pearson test** [102].

Information-theoretic tools – such as typicality [7] – have long been used to analyze the performance of this class of problems. For example, the classical Stein's lemma asserts that asymptotically with the number of data samples approaching infinity [7]

$$\lim_{\alpha \to 0} \lim_{N \to \infty} \frac{1}{N} \cdot \log P_e(\alpha) = -D\left(p_0 || p_1\right) \tag{1.36}$$

where $D\left(\cdot||\cdot\right)$ is the Kullback-Leibler distance between two different distributions.

In Bayesian settings, the performance of a hypothesis testing problem can be described by the **average error probability** given by

$$\begin{aligned} P_e\left(T\right) = \mathbb{P}\left(\mathcal{H}_0\right) \cdot \mathbb{P}\left(T\left(\mathbf{X}^1, \mathbf{X}^2, \ldots, \mathbf{X}^N\right) = \mathcal{H}_1 | \mathcal{H}_0\right) \\ + \mathbb{P}\left(\mathcal{H}_1\right) \cdot \mathbb{P}\left(T\left(\mathbf{X}^1, \mathbf{X}^2, \ldots, \mathbf{X}^N\right) = \mathcal{H}_0 | \mathcal{H}_1\right) \end{aligned} \tag{1.37}$$

where $\mathbb{P}\left(\mathcal{H}_0\right)$ and $\mathbb{P}\left(\mathcal{H}_1\right)$ relate to the prior probabilities ascribed to each of the hypothesis. It is well known that the **maximum *a posteriori* test** (or **maximum *a posteriori* decision rule**) minimizes this average error probability [102].

Information-theoretic tools have also been used to analyze the performance of Bayesian hypothesis testing problems. For example, consider a simple M-ary Bayesian hypothesis testing problem involving M possible hypotheses, which are modelled by a random variable C drawn according to some prior distribution p_C and the data is modelled by a random variable X drawn according to the distribution $p_{X|C}$. In particular, since it is often difficult to characterize in closed-form the minimum average error probability associated with the optimal maximum *a posteriori* test, information-theoretic measures can be used to upper or lower bound this quantity. A lower bound on the minimum average error probability – derived from Fano's inequality – is given by:

$$P_{e,min} = \min_T P_e\left(T\right) \geq 1 - \frac{H\left(C|X\right)}{\log_2\left(M-1\right)}. \tag{1.38}$$

An upper bound on the minimum average error probability is [103]:

$$P_{e,min} = \min_T P_e\left(T\right) \leq 1 - \exp\left(-H\left(C|X\right)\right). \tag{1.39}$$

A number of other bounds on the minimum average error probability involving **Shannon information measures**, **Rényi information measures**, or other generalizations have also been devised over the years [104, 105, 106] that have led to stronger converse results not only in classical information theory problems but also in data science ones [107].

Example: Community detection and estimation on graphs

We now briefly offer examples of hypothesis testing and estimation problems arising in modern data analysis that exhibit sharp statistical and computational

phase transitions which can be revealed using emerging information-theoretic methods.

To add some context, in modern data analysis it is increasingly common for datasets to consist of various items exhibiting complex relationships among them, such as pairwise or multi-way interactions between items. Such datasets can therefore be represented by a **graph** or a **network** of interacting items where the network **vertices** denote different items and the network **edges** denote pairwise interactions between the items. [12] Our example – involving a concrete challenge arising in the analysis of such networks of interacting items – relates to the **detection** and **recovery** of **community** structures within the graph. A community consists of a sub-set of vertices within the graph that are densely connected to one another but sparsely connected to other vertices within the graph [108].

Concretely, consider a simple instance of such problems where one wishes to discern whether the underlying graph is random or whether it contains a dense subgraph (a community). Mathematically, we can proceed by considering two objects: (1) a Erdös-Rényi random graph model $\mathcal{G}(N, q)$ consisting of N vertices where each pair of vertices is connected independently with probability q; and (2) a planted dense subgraph model $\mathcal{G}(N, K, p, q)$ with N vertices where each vertex is assigned to a random set $\mathcal{S}$ with probability K/N ($K \leq N$) and each pair of vertices are connected with probability p if they both are in the set $\mathcal{S}$ and with probability q otherwise ($p > q$). We can then proceed by constructing an hypothesis testing problem where under one hypothesis one postulates that the observed graph is drawn from $\mathcal{G}(N, q)$ and under the other hypothesis one postulates instead that the observed graph is drawn from $\mathcal{G}(N, K, p, q)$. It can then be established in the asymptotic regime $p = cq = O(N^{-\alpha})$, $K = O\left(N^{-\beta}\right)$, $N \to \infty$, that (a) one can detect the community with arbitrarily low error probability with simple linear-time algorithms when $\beta > \frac{1}{2} + \frac{\alpha}{4}$; (b) one can detect the community with arbitrarily low error probability only with no-polynomial-time algorithms when $\alpha < \beta < \frac{1}{2} + \frac{\alpha}{4}$; and (c) there is no test – irrespective of its complexity – that can detect the community with arbitrarily low error probability when $\beta < \min\left(\alpha, \frac{1}{2} + \frac{\alpha}{4}\right)$ [109]. It has also been established that the recovery of the community exhibits identical statistical and computational limits.

This problem in fact falls under a much wider problem class arising in modern data analysis, involving the detection or recovery of structures planted in random objects such as graphs, matrices, or tensors. The characterization of statistical limits in detection or recovery of such structures can be typically done by leveraging various tools: (1) statistical tools such as the first and the second moment methods; (2) information-theoretic methods such as mutual information and rate-distortion; and (3) statistical physics based tools such as the interpola-

[12] Some datasets can also be represented by **hyper-graphs** of interacting items where **vertices** denote the different objects and **hyper-edges** denotes **multi-way** interactions between the different objects.

tion method. In contrast, the characterization of computational limits associated with these statistical problems often involves finding an approximate randomized polynomial-time reduction, mapping certain graph-theoretic problems such as the planted clique problem approximately to the statistical problem under consideration, in order to show that the statistical problem is at least as hard as the planted clique problem. Chapter 13 provides a comprehensive overview of emerging methods – including information-theoretic ones – used to establish both statistical and computational limits in modern data analysis.

1.8.2 Statistical learning

We now survey emering results in statistical learning that are benefiting from information-theoretic methods.

Supervised learning

In the supervised learning set-up one desires to learn a hypothesis based on a set of data examples that can be used to make predictions given new data [90]. In particular, in order to formalize the problem, let $\mathcal{X}$ be the **domain set**, $\mathcal{Y}$ be the **label set**, $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ be the **examples domain**, μ be a distribution on $\mathcal{Z}$, and $\mathcal{W}$ a **hypothesis class** (i.e. $\mathcal{W} = \{W\}$ is a set of **hypotheses** $W : \mathcal{X} \to \mathcal{Y}$). Let also $\mathcal{S} = \left\{\mathbf{z}^1, \dots, \mathbf{z}^N\right\} = \left\{(\mathbf{x}^1, \mathbf{y}^1), \dots, (\mathbf{x}^N, \mathbf{y}^N)\right\} \in \mathcal{Z}^N$ be the **training set** – consisting of a number of data points and their associated labels – drawn i.i.d. from $\mathcal{Z}$ according to μ. A **learning algorithm** is a Markov kernel that maps the **training set** $\mathcal{S}$ to an element W of the hypothesis class $\mathcal{W}$ per the probability law $p_{W|\mathcal{S}}$.

A key challenge relates to understanding the generalization ability of the learning algorithm, where the **generalization error** corresponds to the difference between the **expected (or true) error** and the **training (or empirical) error**. In particular, by considering a non-negative **loss function** $L : \mathcal{W} \times \mathcal{Z} \to \mathbb{R}^+$, one can define the expected error and the training error associated with a hypothesis W as follows:

$$\mathsf{loss}_\mu(W) = \mathbb{E}\{L(W, \mathcal{Z})\} \qquad \text{and} \qquad \mathsf{loss}_\mathcal{S}(W) = \frac{1}{N}\sum_{i=1}^{N} L(W, \mathbf{z}^i)$$

respectively. The generalization error is given by

$$\mathsf{gen}(\mu, W) = \mathsf{loss}_\mu(W) - \mathsf{loss}_\mathcal{S}(W)$$

and its expected value is given by

$$\mathsf{gen}(\mu, p_{W|\mathcal{S}}) = \mathbb{E}\left\{\mathsf{loss}_\mu(W) - \mathsf{loss}_\mathcal{S}(W)\right\}$$

where the expectation is with respect to the joint distribution of the algorithm input (the training set) and the algorithm output (the hypothesis).

A number of approaches have been developed throughout the years to characterize the generalization error of a learning algorithm, relying on either certain

complexity measures of the hypothesis space or certain properties of the learning algorithm. These include VC-based bounds [110], algorithmic stability based bounds [111], algorithmic robustness based bounds [112], PAC-Bayesian bounds [113], and many more. However, many of these generalization error bounds cannot explain the generalization abilities of a variety of machine learning methods for various reasons: (1) some of the bounds depend only on the hypothesis class and not on the learning algorithm, (2) existing bounds do not easily exploit dependences between different hypothesis, and (3) existing bounds also do not exploit dependences between the learning algorithm input and output.

More recently, approaches leveraging information-theoretic tools have been emerging to characterize the generalization ability of various learning methods. Such approaches often express the generalization error in terms of certain information measures between the algorithm input (the training dataset) and the algorithm output (the hypothesis), thereby incorporating the various ingredients associated with the learning problem, including the dataset distribution, the hypothesis space, and the learning algorithm itself. In particular, inspired by [114], Xu and Raginsky [115] derive an upper bound on the generalization error, applicable to σ-subgaussian loss functions, given by:

$$\left|\mathsf{gen}(\mu, p_{W|\mathcal{S}})\right| \leq \sqrt{\frac{2\sigma^2}{N} \cdot I(\mathcal{S}; W)}$$

where $I(\mathcal{S}; W)$ corresponds to the mutual information between the input – the dataset – and the output – the hypothesis – of the algorithm. This bound supports the intuition that the less information the output of the algorithm contains about the input to the algorithm the less it will overfit, providing a means to strike a balance between the ability to fit data and the ability to generalize to new data by controlling the algorithm's input-output mutual information. Raginsky et al. [116] also propose similar upper bounds on the generalization error based on several information-theoretic measures of algorithmic stability, capturing the idea that the output of a stable learning algorithm cannot depend "too much" on any particular training example. Other generalization error bounds involving information-theoretic quantities appear in [117, 118]. In particular, Asadi et al. [118] combine chaining and mutual information methods to derive generalization error bounds that significantly outperform existing ones.

Of particular relevance, these information-theoretic based generalization error bounds have also been used to cast further insight onto machine learning models and algorithms. For example, Pensia et al. [119] build upon the work by Xu and Raginsky [115] to derive very general generalization error bounds for a broad class of iterative algorithms that are characterized by bounded, noisy updates with Markovian structure, including stochastic gradient Langevin dynamics (SGLD) and variants of the stochastic gradient Hamiltonian Monte Carlo (SGHMC) algorithm. This work demonstrates that mutual information is a very effective tool for bounding the generalization error of a large class of iterative **empirical risk minimization** (ERM) algorithms. Zhang et al. [120], on the other

hand, build upon the work by Xu and Raginsky [115] to study the expected generalization error of deep neural networks, and offer a bound that shows that the error decreases exponentially to zero with the increase of convolutional and pooling layers in the network. Other works that study the generalization ability of deep networks based on information-theoretic considerations and measures include [121, 122]. Chapters 10 and 11 scope these directions in supervised learning problems.

Unsupervised learning

In unsupervised learning set-ups, one desires instead to understand the structure associated with a set of data examples. In particular, multivariate information-theoretic functionals such as partition information, minimum partition information, and multiinformation have been recently used in the formulation of unsupervised clustering problems [123, 124]. Chapter 9 elaborates further on such approaches to unsupervised learning problems.

1.8.3 Distributed inference and learning

Finally, we add that there has also been considerable interest in the generalization of the classical statistical inference and learning problems overviewed here to the distributed setting, where a statistician / learner only has access to data distributed across various terminals via a series of limited-capacity channels. In particular, much progress has been made in distributed estimation [125, 126, 127], hypothesis testing [128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138], learning [139, 140], and function computation [141] problems in recent years. Chapters 14 and 15 elaborate further on how information theory is advancing the state-of-the-art for this class of problems.

1.9 Discussion and Conclusion

This chapter overviewed the classical information-theoretic problems of data compression and communication, questions arising within the context of these problems, and classical information-theoretic tools used to illuminate fundamental architectures, schemes, and limits for data compression and communication.

We then discussed how information-theoretic methods are currently advancing the frontier of data science by unveiling new data processing architectures, data processing limits, and algorithms. In particular, we scoped how information theory is leading to a new understanding of data acquisition architectures, and provided an overview of how information-theoretic methods have been uncovering limits and algorithms for linear and nonlinear representation learning problems, including deep learning. Finally, we also overviewed how information-theoretic tools have been contributing to our understanding of limits and algorithms for statistical inference and learning problems.

Beyond the typical data acquisition, data representation, and data analysis tasks covered throughout this introduction, there are also various other emerging challenges in data science that are benefiting from information-theoretic techniques. For example, **privacy** is becoming a very relevant research area in data science in view of the fact that data analysis can not only reveal useful insights but also potentially disclose sensitive information about individuals. Differential privacy is an inherently information-theoretic framework that can be used as the basis for the development of data release mechanisms that control the amount of private information leaked to a data analyst while retaining some degree of utility [142]. Other information-theoretic frameworks have also been used to develop data pre-processing mechanisms that strike a balance between the amount of useful information and private information leaked to a data analyst (e.g. [143]).

Fairness is likewise becoming a very relevant area in data science because data analysis can also potentially exacerbate biases in decision making, such as discriminatory treatments of individuals based on membership of a legally protected group such as race or gender. Such biases may arise when protected variables (or correlated ones) are used explicitly in the decision making. Biases also arise when learning algorithms inheriting possible biases present in training sets are used in decision making. Recent works have concentrated on the development of information-theoretic based data pre-processing schemes that aim to simultaneously control discrimination as well as preserve utility (e.g. [144]).

Overall, we anticipate that information-theoretic methods will play an increasingly important role in our understanding of data science in upcoming years, including in shaping data processing architectures, in revealing fundamental data processing limits, and in the analysis, design, and optimization of new data processing algorithms.

References

[1] C. E. Shannon, "A mathematical theory of communications," *Bell System Technical Journal*, vol. 27, no. 3–4, pp. 379–423, 623–656, Jul.–Oct. 1948.

[2] R. G. Gallager, *Information theory and reliable communications.* Wiley, 1968.

[3] T. Berger, *Rate distortion theory: A mathematical basis for data compression.* Prentice-Hall, 1971.

[4] I. Csiszár and J. Körner, *Information theory: Coding theorems for discrete memoryless systems.* Cambridge University Press, 2011.

[5] A. Gersho and R. M. Gray, *Vector quantization and signal compression.* Kluwer Academic Publishers, 1991.

[6] D. J. C. MacKay, *Information theory, inference and learning algorithms.* Cambridge University Press, 2003.

[7] T. M. Cover and J. A. Thomas, *Elements of information theory.* John Wiley & Sons, 2006.

[8] R. W. Yeung, *Information theory and network coding.* Springer, 2008.

[9] A. El Gamal and Y.-H. Kim, *Network information theory.* Cambridge University Press, 2011.

[10] E. Arikan, "Some remarks on the nature of the cutoff rate," in *Proc. Workshop Information Theory and Applications (ITA'06)*, Feb. 2006.

[11] R. E. Blahut, *Theory and practice of error control codes.* Addison-Wesley Publishing Company, 1983.

[12] S. Lin and D. J. Costello, *Error control coding.* Pearson, 2005.

[13] R. M. Roth, *Introduction to coding theory.* Cambridge University Press, 2006.

[14] T. Richardson and R. Urbanke, *Modern coding theory.* Cambridge University Press, 2008.

[15] W. E. Ryan and S. Lin, *Channel codes: Classical and modern.* Cambridge University Press, 2009.

[16] E. Arikan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Trans. Inform. Theory*, vol. 55, no. 7, pp. 3051–3073, Jul. 2009.

[17] A. Jimeńez-Feltström and K. S. Zigangirov, "Time-varying periodic convolutional codes with low-density parity-check matrix," *IEEE Trans. Inform. Theory*, vol. 45, no. 2, pp. 2181–2191, 1999.

[18] M. Lentmaier, A. Sridharan, D. J. J. Costello, and K. S. Zigangirov, "Iterative decoding threshold analysis for LDPC convolutional codes," *IEEE Trans. Inform. Theory*, vol. 56, no. 10, pp. 5274–5289, 2010.

[19] S. Kudekar, T. J. Richardson, and R. L. Urbanke, "Threshold saturation via spatial coupling: Why convolutional LDPC ensembles perform so well over the BEC," *IEEE Trans. Inform. Theory*, vol. 57, no. 2, pp. 803–834, 2011.

[20] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Mag.*, vol. 25, no. 2, pp. 21–30, 2008.

[21] H. Q. Ngo and D.-Z. Du, "A survey on combinatorial group testing algorithms with applications to DNA library screening," *Discrete Math. Problems with Medical Appl.*, vol. 55, pp. 171–182, 2000.

[22] G. K. Atia and V. Saligrama, "Boolean compressed sensing and noisy group testing," *IEEE Trans. Inform. Theory*, vol. 58, no. 3, pp. 1880–1901, 2012.

[23] D. Donoho and J. Tanner, "Observed universality of phase transitions in high-dimensional geometry, with implications for modern data analysis and signal processing," *Phil. Trans. Royal Soc. A: Mathematical, Physical and Engineering Sciences*, pp. 4273–4293, 2009.

[24] D. Amelunxen, M. Lotz, M. B. McCoy, and J. A. Tropp, "Living on the edge: Phase transitions in convex programs with random data," *Information and Inference*, vol. 3, no. 3, pp. 224–294, 2014.

[25] J. Banks, C. Moore, R. Vershynin, N. Verzelen, and J. Xu, "Information-theoretic bounds and phase transitions in clustering, sparse PCA, and submatrix localization," *IEEE Trans. Inform. Theory*, vol. 64, no. 7, pp. 4872–4894, 2018.

[26] R. Monasson, R. Zecchina, S. Kirkpatrick, B. Selman, and L. Troyansky, "Determining computational complexity from characteristic 'phase transitions'," *Nature*, vol. 400, no. 6740, pp. 133–137, 1999.

[27] G. Zeng and Y. Lu, "Survey on computational complexity with phase transitions and extremal optimization," in *Proc. 48th IEEE Conf. Decision and Control (CDC'09)*, Dec 2009, pp. 4352–4359.

[28] Y. C. Eldar, *Sampling theory: Beyond bandlimited systems*. Cambridge University Press, 2014.

[29] C. E. Shannon, "Coding theorems for a discrete source with a fidelity criterion," *IRE National Convention Record, Part 4*, pp. 142–163, 1959.

[30] A. Kipnis, A. J. Goldsmith, Y. C. Eldar, and T. Weissman, "Distortion-rate function of sub-nyquist sampled gaussian sources," *IEEE Trans. Inform. Theory*, vol. 62, no. 1, pp. 401–429, 2016.

[31] A. Kipnis, Y. C. Eldar, and A. J. Goldsmith, "Analog-to-digital compression: A new paradigm for converting signals to bits," *IEEE Signal Processing Mag.*, vol. 35, no. 3, pp. 16–39, 2018.

[32] ——, "Fundamental distortion limits of analog-to-digital compression," *IEEE Trans. Inform. Theory*, vol. 64, no. 9, pp. 6013–6033, 2018.

[33] M. R. D. Rodrigues, N. Deligiannis, L. Lai, and Y. C. Eldar, "Rate-distortion trade-offs in acquisition of signal parameters," in *Proc. IEEE Intl. Conf. Acoust. Speech Signal Process. (ICASSP'17)*, New Orleans, LA, March 2017.

[34] N. Shlezinger, Y. C. Eldar, and M. R. D. Rodrigues, "Hardware-limited task-based quantization," *submitted to IEEE Trans. Signal Processing*, 2018.

[35] ——, "Asymptotic task-based quantization with application to massive mimo," *submitted to IEEE Trans. Signal Processing*, 2018.

[36] A. Argyriou, T. Evgeniou, and M. Pontil, "Convex multi-task feature learning," *Mach. Learn.*, vol. 73, no. 3, pp. 243–272, 2008.

[37] A. Coates, A. Ng, and H. Lee, "An analysis of single-layer networks in unsupervised feature learning," in *Proc. 14th Intl. Conf. Artificial Intelligence and Statistics (AISTATS'11)*, Fort Lauderdale, FL, Apr. 2011, pp. 215–223.

[38] I. Tosic and P. Frossard, "Dictionary learning," *IEEE Signal Processing Mag.*, vol. 28, no. 2, pp. 27–38, Mar. 2011.

[39] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, 2013.

[40] S. Yu, K. Yu, V. Tresp, H.-P. Kriegel, and M. Wu, "Supervised probabilistic principal component analysis," in *Proc. 12th ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining (KDD'06)*, Aug. 2006, pp. 464–473.

[41] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Supervised dictionary learning," in *Proc. Advances in Neural Information Processing Systems (NeurIPS'09)*, D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, Eds., Vancouver, BC, Canada, Dec. 2009, pp. 1033–1040.

[42] V. Vu and J. Lei, "Minimax rates of estimation for sparse PCA in high dimensions," in *Proc. 15th Intl. Conf. Artificial Intelligence and Statistics (AISTATS'12)*, La Palma, Canary Islands, Apr. 2012, pp. 1278–1286.

[43] T. T. Cai, Z. Ma, and Y. Wu, "Sparse PCA: Optimal rates and adaptive estimation," *Annals Statist.*, vol. 41, no. 6, pp. 3074–3110, 2013.

[44] A. Jung, Y. C. Eldar, and N. Görtz, "On the minimax risk of dictionary learning," *IEEE Trans. Inform. Theory*, vol. 62, no. 3, pp. 1501–1515, 2016.

[45] Z. Shakeri, W. U. Bajwa, and A. D. Sarwate, "Minimax lower bounds on dictionary learning for tensor data," *IEEE Trans. Inform. Theory*, vol. 64, no. 4, 2018.

[46] H. Hotelling, "Analysis of a complex of statistical variables into principal components," *J. Edu. Psych.*, vol. 6, no. 24, pp. 417–441, 1933.

[47] M. E. Tipping and C. M. Bishop, "Probabilistic principal component analysis," *J. Roy. Statist. Soc. Ser. B*, vol. 61, no. 3, pp. 611–622, 1999.

[48] I. T. Jolliffe, *Principal component analysis*, 2nd ed. Springer-Verlag, 2002.

[49] P. Comon, "Independent component analysis, A new concept?" *Signal Processing*, vol. 36, no. 3, pp. 287–314, 1994.

[50] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent component analysis*. John Wiley & Sons, 2004.

[51] P. Belhumeur, J. Hespanha, and D. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intel.*, vol. 19, no. 7, pp. 711–720, 1997.

[52] J. Ye, R. Janardan, and Q. Li, "Two-dimensional linear discriminant analysis," in *Proc. Advances in Neural Information Processing Systems (NeurIPS'04)*, Vancouver, BC, Canada, Dec. 2004, pp. 1569–1576.

[53] T. Hastie, R. Tibshirani, and J. Friedman, *The elements of statistical learning: Data mining, inference, and prediction*, 2nd ed. Springer, 2016, no. 10.

[54] A. Hyvarinen, "Fast and robust fixed-point algorithms for independent component analysis," *IEEE Trans. Neural Networks*, vol. 10, no. 3, pp. 626–634, 1999.

[55] D. Erdogmus, K. E. Hild, Y. N. Rao, and J. C. Príncipe, "Minimax mutual information approach for independent component analysis," *Neural Comput.*, vol. 16, no. 6, pp. 1235–1252, 2004.

[56] A. Birnbaum, I. M. Johnstone, B. Nadler, and D. Paul, "Minimax bounds for sparse PCA with noisy high-dimensional data," *Ann. Statist.*, vol. 41, no. 3, pp. 1055–1084, 2013.

[57] R. Krauthgamer, B. Nadler, and D. Vilenchik, "Do semidefinite relaxations solve sparse PCA up to the information limit?" *Ann. Statist.*, vol. 43, no. 3, pp. 1300–1322, 2015.

[58] Q. Berthet and P. Rigollet, "Representation learning: A review and new perspectives," *Ann. Statist.*, vol. 41, no. 4, pp. 1780–1815, 2013.

[59] T. Cai, Z. Ma, and Y. Wu, "Optimal estimation and rank detection for sparse spiked covariance matrices," *Probab. Theory Relat. Fields*, vol. 161, no. 3–4, pp. 781–815, 2015.

[60] A. Onatski, M. Moreira, and M. Hallin, "Asymptotic power of sphericity tests for high-dimensional data," *Ann. Statist.*, vol. 41, no. 3, pp. 1204–1231, 2013.

[61] A. Perry, A. Wein, A. Bandeira, and A. Moitra, "Optimality and sub-optimality of PCA for spiked random matrices and synchronization," *arXiv:1609.05573*, 2016.

[62] Z. Ke, "Detecting rare and weak spikes in large covariance matrices," *arXiv:1609.00883*, 2018.

[63] D. L. Donoho and C. Grimes, "Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data," *Proc. Natl. Acad. Sci.*, vol. 100, no. 10, pp. 5591–5596, May 2003.

[64] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[65] R. Jenssen, "Kernel entropy component analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 5, pp. 847–860, May 2010.

[66] B. Schölkopf, A. Smola, and K.-R. Müller, "Kernel principal component analysis," in *Proc. Intl. Conf. Artificial Neural Networks (ICANN'97)*, Lausanne, Switzerland, Oct. 1997, pp. 583–588.

[67] J. Yang, X. Gao, D. Zhang, and J. yu Yang, "Kernel ICA: An alternative formulation and its application to face recognition," *Pattern Recognit.*, vol. 38, no. 10, pp. 1784 – 1787, 2005.

[68] S. Mika, G. Ratsch, J. Weston, B. Scholkopf, and K. R. Mullers, "Fisher discriminant analysis with kernels," in *Proc. IEEE Workshop Neural Networks for Signal Processing IX*, Aug 1999, pp. 41–48.

[69] H. Narayanan and S. Mitter, "Sample complexity of testing the manifold hypothesis," in *Proc. Advances in Neural Information Processing Systems (NeurIPS'10)*, Vancouver, BC, Canada, Dec. 2010, pp. 1786–1794.

[70] K. Kreutz-Delgado, J. F. Murray, B. D. Rao, K. Engan, T.-W. Lee, and T. J. Sejnowski, "Dictionary learning algorithms for sparse representation," *Neural Comput.*, vol. 15, no. 2, pp. 349–396, 2003.

[71] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

[72] Q. Zhang and B. Li, "Discriminative K-SVD for dictionary learning in face recognition," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR'10)*, San Francisco, CA, USA, Jun. 2010, pp. 2691–2698.

[73] Q. Geng and J. Wright, "On the local correctness of ℓ^1-minimization for dictionary learning," in *Proc. IEEE Intl. Symp. Information Theory (ISIT'14)*, Honolulu, HI, USA, June 2014, pp. 3180–3184.

[74] A. Agarwal, A. Anandkumar, P. Jain, P. Netrapalli, and R. Tandon, "Learning sparsely used overcomplete dictionaries," in *Proc. 27th Conf. Learning Theory (COLT'14)*, Barcelona, Spain, Jun. 2014, pp. 123–137.

[75] S. Arora, R. Ge, and A. Moitra, "New algorithms for learning incoherent and overcomplete dictionaries," in *Proc. 27th Conf. Learning Theory (COLT'14)*, Barcelona, Spain, Jun. 2014, pp. 779–806.

[76] R. Gribonval, R. Jenatton, and F. Bach, "Sparse and spurious: dictionary learning with noise and outliers," *IEEE Trans. Inform. Theory*, vol. 61, no. 11, pp. 6298–6319, 2015.

[77] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Proc. Advances in Neural Information Processing Systems 13 (NeurIPS'01)*, 2001, pp. 556–562.

[78] A. Cichocki, R. Zdunek, A. H. Phan, and S.-i. Amari, *Nonnegative matrix and tensor factorizations: Applications to exploratory multi-way data analysis and blind source separation*. John Wiley & Sons, 2009.

[79] M. Alsan, Z. Liu, and V. Y. F. Tan, "Minimax lower bounds for nonnegative matrix factorization," in *Proc. IEEE Statistical Signal Processing Workshop (SSP'18)*, Freiburg, Germany, June 2018, pp. 363–367.

[80] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436—-444, 2015.

[81] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[82] N. Tishby and N. Zaslavsky, "Deep learning and the information bottleneck principle," in *Proc. IEEE Information Theory Workshop (ITW'15)*, Jeju Island, South Korea, Apr.–May 2015.

[83] R. Shwartz-Ziv and N. Tishby, "Opening the black box of deep neural networks via information," *arXiv:1703.00810*, 2017.

[84] C. W. Huang and S. S. Narayanan, "Flow of rényi information in deep neural networks," in *Proc. IEEE Intl. Workshop Machine Learning for Signal Processing (MLSP'16)*, Vietri sul Mare, Italy, Sep.

[85] P. Khadivi, R. Tandon, and N. Ramakrishnan, "Flow of information in feed-forward deep neural networks," *arXiv:1603.06220*, 2016.

[86] S. Yu, R. Jenssen, and J. Príncipe, "Understanding convolutional neural network training with information theory," *arXiv:1804.09060*, 2018.

[87] S. Yu and J. Príncipe, "Understanding autoencoders with information theoretic concepts," *arXiv:1804.00057*, 2018.

[88] A. Achille and S. Soatto, "Emergence of invariance and disentangling in deep representations," *arXiv:1706.01350*, 2017.

[89] R. D. Hjelm, A. Fedorov, S. Lavoie-Marchildon, K. Grewal, P. Bachman, A. Trischler, and Y. Bengio, "Learning deep representations by mutual information estimation and maximization," in *Intl. Conf. Learning Representations (ICLR'19)*, New Orleans, LA, USA, May 2019.

[90] S. Shalev-Shwartz and S. Ben-David, *Understanding machine learning: From theory to algorithms*. Cambridge University Press, 2014.

[91] H. Akaike, "A new look at the statistical model identification," *IEEE Trans. Automat. Control*, vol. 19, no. 6, pp. 716–723, 1974.

[92] A. Barron, J. Rissanen, and B. Yu, "The minimum description length principle in coding and modeling," *IEEE Trans. Inform. Theory*, vol. 44, no. 6, pp. 2743–2760, 1998.

[93] M. J. Wainwright, "Information-theoretic limits on sparsity recovery in the high-dimensional and noisy setting," *IEEE Trans. Inform. Theory*, vol. 55, no. 12, pp. 5728–5741, 2009.

[94] ——, "Sharp thresholds for high-dimensional and noisy sparsity recovery using ℓ_1-constrained quadratic programming (lasso)," *IEEE Trans. Inform. Theory*, vol. 55, no. 5, pp. 2183–2202, 2009.

[95] G. Raskutti, M. J. Wainwright, and B. Yu, "Minimax rates of estimation for high-dimensional linear regression overℓ_q-balls," *IEEE Trans. Inform. Theory*, vol. 57, no. 10, pp. 6976–6994, 2011.

[96] D. Guo, S. Shamai, and S. Verdú, "Mutual information and minimum mean-square error in gaussian channels," *IEEE Trans. Inform. Theory*, vol. 51, no. 4, pp. 1261–1282, 2005.

[97] ——, "Mutual information and conditional mean estimation in poisson channels," *IEEE Trans. Inform. Theory*, vol. 54, no. 5, pp. 1837–1849, 2008.

[98] A. Lozano, A. M. Tulino, and S. Verdú, "Optimum power allocation for parallel gaussian channels with arbitrary input distributions," *IEEE Trans. Inform. Theory*, vol. 52, no. 7, pp. 3033–3051, 2006.

[99] F. Pérez-Cruz, M. R. D. Rodrigues, and S. Verdú, "Multiple-antenna fading channels with arbitrary inputs: Characterization and optimization of the information rate," *IEEE Trans. Inform. Theory*, vol. 56, no. 3, pp. 1070–1084, 2010.

[100] M. R. D. Rodrigues, "Multiple-antenna fading channels with arbitrary inputs: Characterization and optimization of the information rate," *IEEE Trans. Inform. Theory*, vol. 60, no. 1, pp. 569–585, 2014.

[101] A. G. C. P. Ramos and M. R. D. Rodrigues, "Fading channels with arbitrary inputs: Asymptotics of the constrained capacity and information and estimation measures," *IEEE Trans. Inform. Theory*, vol. 60, no. 9, pp. 5653–5672, 2014.

[102] S. M. Kay, *Fundamentals of statistical signal processing: Detection theory*. Prentice Hall, 1998.

[103] M. Feder and N. Merhav, "Relations between entropy and error probability," *IEEE Trans. Inform. Theory*, vol. 40, no. 1, pp. 259–266, 1994.

[104] I. Sason and S. Verdú, "Arimoto–rényi conditional entropy and bayesian m-ary hypothesis testing," *IEEE Trans. Inform. Theory*, vol. 64, no. 1, pp. 4–25, 2018.

[105] Y. Polyanskiy, H. V. Poor, and S. Verdú, "Channel coding rate in the finite blocklength regime," *IEEE Trans. Inform. Theory*, vol. 56, no. 5, pp. 2307–2359, 2010.

[106] G. Vazquez-Vilar, A. T. Campo, A. Guillén i Fábregas, and A. Martinez, "Bayesian m-ary hypothesis testing: The meta-converse and verdú-han bounds are tight," *IEEE Trans. Inform. Theory*, vol. 62, no. 5, pp. 2324–2333, 2016.

[107] R. Venkataramanan and O. Johnson, "A strong converse bound for multiple hypothesis testing, with applications to high-dimensional estimation," *Electron. J. Stat.*, vol. 12, no. 1, pp. 1126–1149, 2018.

[108] E. Abbe, "Community detection and stochastic block models: Recent developments," *J. Mach. Learn. Res.*, vol. 18, pp. 1–86, 2018.

[109] B. Hajek, Y. Wu, and J. Xu, "Computational lower bounds for community detection on random graphs," in *Proc. 28th Conf. Learning Theory (COLT'15)*, Paris, France, Jul. 2015, pp. 1–30.

[110] V. N. Vapnik, "An overview of statistical learning theory," *IEEE Trans. Neural Netw.*, vol. 10, no. 5, pp. 988–999, 1999.

[111] O. Bousquet and A. Elisseeff, "Stability and generalization," *J. Mach. Learn. Res.*, vol. 2, pp. 499–526, 2002.

[112] H. Xu and S. Mannor, "Robustness and generalization," *Mach. Learn.*, vol. 86, no. 3, pp. 391–423, 2012.

[113] D. A. McAllester, "Pac-bayesian stochastic model selection," *Mach. Learn.*, vol. 51, pp. 5–21, 2003.

[114] D. Russo and J. Zou, "How much does your data exploration overfit? controlling bias via information usage," *arXiv:1511.05219*, 2016.

[115] A. Xu and M. Raginsky, "Information-theoretic analysis of generalization capability of learning algorithms," in *Proc. Advances in Neural Information Processing Systems (NeurIPS'17)*, Long Beach, CA, USA, Dec. 2017.

[116] M. Raginsky, A. Rakhlin, M. Tsao, Y. Wu, and A. Xu, "Information-theoretic analysis of stability and bias of learning algorithms," in *Proc. IEEE Information Theory Workshop (ITW'16)*, Cambridge, UK, Sep. 2016.

[117] R. Bassily, S. Moran, I. Nachum, J. Shafer, and A. Yehudayof, "Learners that use little information," *arXiv:1710.05233*, 2018.

[118] A. R. Asadi, E. Abbe, and S. Verdú, "Chaining mutual information and tightening generalization bounds," *arXiv:1806.03803*, 2018.

[119] A. Pensia, V. Jog, and P. L. Loh, "Generalization error bounds for noisy, iterative algorithms," *arXiv:1801.04295v1*, 2018.

[120] J. Zhang, T. Liu, and D. Tao, "An information-theoretic view for deep learning," *arXiv:1804.09060*, 2018.

[121] M. Vera, P. Piantanida, and L. R. Vega, "The role of information complexity and randomization in representation learning," *arXiv:1802.05355*, 2018.

[122] M. Vera, L. R. Vega, and P. Piantanida, "Compression-based regularization with an application to multi-task learning," *arXiv:1711.07099*, 2018.

[123] C. Chan, A. Al-Bashadsheh, and Q. Zhou, "Info-clustering: A mathematical theory of data clustering," *IEEE Trans. Mol. Biol. Multi-Scale Commun.*, vol. 2, no. 1, pp. 64–91, 2016.

[124] R. K. Raman and L. R. Varshney, "Universal joint image clustering and registration using multivariate information measures," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 5, pp. 928–943, 2018.

[125] Z. Zhang and T. Berger, "Estimation via compressed information," *IEEE Trans. Inform. Theory*, vol. 34, no. 2, pp. 198—-211, 1988.

[126] T. S. Han and S. Amari, "Parameter estimation with multiterminal data compression," *IEEE Trans. Inform. Theory*, vol. 41, no. 6, pp. 1802—-1833, 1995.

[127] Y. Zhang, J. C. Duchi, M. I. Jordan, and M. J. Wainwright, "Information-theoretic lower bounds for distributed statistical estimation with communication constraints," in *Proc. Advances in Neural Information Processing Systems (NeurIPS'13)*, Lake Tahoe, CA, USA, Dec. 2013.

[128] R. Ahlswede and I. Csiszár, "Hypothesis testing with communication constraints," *IEEE Trans. Inform. Theory*, vol. 32, no. 4, pp. 533–542, 1986.

[129] T. S. Han, "Hypothesis testing with multiterminal data compression," *IEEE Trans. Inform. Theory*, vol. 33, no. 6, pp. 759–772, 1987.

[130] T. S. Han and K. Kobayashi, "Exponential-type error probabilities for multiterminal hypothesis testing," *IEEE Trans. Inform. Theory*, vol. 35, no. 1, pp. 2–14, 1989.

[131] T. S. Han and S. Amari, "Statistical inference under multiterminal data compression," *IEEE Trans. Inform. Theory*, vol. 44, no. 6, pp. 2300–2324, 1998.

[132] H. M. H. Shalaby and A. Papamarcou, "Multiterminal detection with zero-rate data compression," *IEEE Trans. Inform. Theory*, vol. 38, no. 2, pp. 254–267, 1992.

[133] G. Katz, P. Piantanida, R. Couillet, and M. Debbah, "On the necessity of binning for the distributed hypothesis testing problem," in *Proc. IEEE Intl. Symp. Information Theory (ISIT'15)*, Hong Kong, China, Jun. 2015.

[134] Y. Xiang and Y. Kim, "Interactive hypothesis testing against independence," in *Proc. IEEE Intl. Symposium Information Theory (ISIT'13))*, Istanbul, Turkey, Jul. 2013.

[135] W. Zhao and L. Lai, "Distributed testing against independence with conferencing encoders," in *Proc. IEEE Information Theory Workshop (ITW'15))*, Jeju Island, South Korea, Oct. 2015.

[136] ——, "Distributed testing with zero-rate compression," in *Proc. IEEE Intl. Symp. Information Theory (ISIT'15)*, Hong Kong, China, Jun. 2015.

[137] ——, "Distributed detection with vector quantizer," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 2, no. 2, pp. 105–119, 2016.

[138] ——, "Distributed testing with cascaded encoders," *IEEE Trans. Inform. Theory*, vol. 64, no. 11, pp. 7339–7348, 2018.

[139] M. Raginsky, "Learning from compressed observations," in *Proc. IEEE Information Theory Workshop (ITW'07)*, Lake Tahoe, CA, USA, Sep. 2007.

[140] ——, "Achievability results for statistical learning under communication constraints," in *Proc. IEEE Intl. Symp. Information Theory (ISIT'09)*, Seoul, South Korea, June-July 2009.

[141] A. Xu and M. Raginsky, "Information-theoretic lower bounds for distributed function computation," *IEEE Trans. Inform. Theory*, vol. 63, no. 4, pp. 2314–2337, 2017.

[142] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Foundations and Trends® in Theoretical Computer Science*, vol. 9, no. 3–4, pp. 211–407, 2014.

[143] J. Liao, L. Sankar, V. Y. F. Tan, and F. P. Calmon, "Hypothesis testing under mutual information privacy constraints in the high privacy regime," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 4, pp. 1058–1071, 2018.

[144] F. P. Calmon, D. Wei, B. Vinzamuri, K. N. Ramamurthy, and K. R. Varshney, "Data pre-processing for discrimination prevention: Information-theoretic optimization and analysis," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 5, pp. 1106–1119, 2018.

Index