Faculty of Science · Department of Computer Science · Group of Prof. Robert Bamler

Information Theory With Applications to Data Compression

Robert Bamler · Tutorial at IMPRS-IS Boot Camp 2024

While you're waiting:
If you brought a laptop (optional), please go to `https://bamler-lab.github.io/bootcamp24` and test if you can run the linked Google Colab notebook. You can also find the slides at this link.

Let's Debate

Slides and code available at:
`https://bamler-lab.github.io/bootcamp24`

1. **Which of the following two messages contains more information?**
   (a) "The instructor of this tutorial knows how to solve a quadratic equation."
   ✓ longer; ✗ not much *new* information (little *surprise*); ✓ *useful* to judge my qualification.
   (b) "The instructor of this tutorial likes roller coasters."
   ✓ *new* information; ✗ do you really care?
2. **Which of the following two pairs of quantities are more strongly correlated:**
   (a) the *volumes* and *radii* of (spherical) glass marbles (of random sizes and colors)
   ✓ *exact* correspondence: $V = \frac{4}{3}\pi r^3$, so once we know r, telling us V gives us no *new* information.
   ✗ *nonlinear* relation $\Longrightarrow$ lower Pearson's correlation coefficient (see code).
   (b) the *volumes* and *masses* of glass marbles (of random sizes and colors)
   ✓ *linear* relation: $m = \rho V$, so m and V essentially convey the same information in different units;
   ✗ not an exact correspondence: density ρ varies slightly depending on color;
   $\Longrightarrow$ even if we know V, we can still learn some *new* information by measuring m and vice versa.

So, What is Information Theory?

Information theory provides tools to analyze:

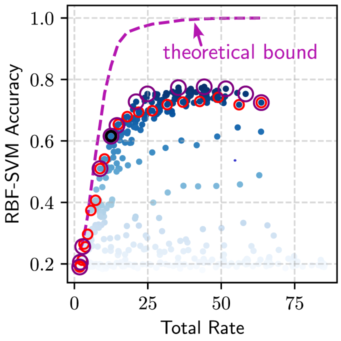
- the *quantity* (i.e., amount) of information in some data;
- more precisely, the amount of *novelty/surprisingness* of a piece of information w.r.t.:
  (a) prior beliefs (e.g., an ML researcher probably knows high-school math); or
  (b) a different piece of information (when quantifying correlations).

Information theory is oblivious to:

- the *quality* of a piece of information (e.g., its utility, urgency, or even truthfulness).
- how a piece of information is represented in the data, e.g.,
  - the volume and radius of a sphere are different representations of the same piece information;
  - for a given neural network with known weights, its output cannot contain more information than its input. $\Rightarrow$ Inf. theory can provide upper bounds, e.g., on how much useful information an optimal model can extract from some latent representation.
- computational costs: compressed representations of the same information are sometimes easier but often *harder to process* than their uncompressed counterparts. ⚠

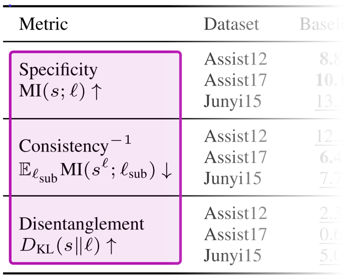
Where Are These Tools Useful?

Theoretical bounds for model performance



[Tim Xiao, RB, ICLR 2023]

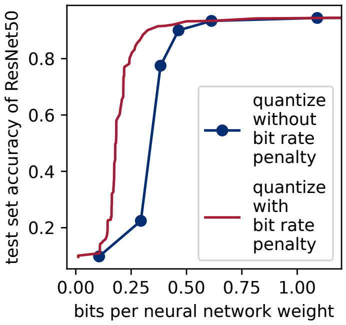
Analyze abstract representation vectors



Metric	Dataset	Baseli
Specificity $\mathrm{MI}(s;\ell)\uparrow$	Assist12	8.8
	Assist17	10.
	Junyi15	13.
Consistency^{-1} $\mathbb{E}_{\ell_{\mathrm{sub}}}\mathrm{MI}(s^{\ell};\ell_{\mathrm{sub}})\downarrow$	Assist12	12.
	Assist17	6.4
	Junyi15	7.7
Disentanglement $D_{\mathrm{KL}}(s\|\ell)\uparrow$	Assist12	2.3
	Assist17	0.6
	Junyi15	5.0

[Hanqi Zhou, RB, C. M. Wu, Á. Tejero-Cantero, ICLR 2024]

Data Compression (“Source Coding”)



[Alexander Conzelmann, RB; coming soon]

My Promise for This Tutorial

entropy

$H(P) = \mathbb{E}_P[-\log P(X)]$

Kullback-Leibler divergence

$D_{\mathrm{KL}}(P\|Q) = -\int p(x)\log\frac{q(x)}{p(x)}\,dx$

cross entropy

$H(P,Q) = \mathbb{E}_P[-\log Q(X)]$

mutual information

$I_P(X;Y) = -\int\!\!\int p(x,y)\log\frac{p(x)p(y)}{p(x,y)}\,dx\,dy$

Why?

What for?

Quantifying Information

[Shannon, *A Mathematical Theory of Communication*, 1948]

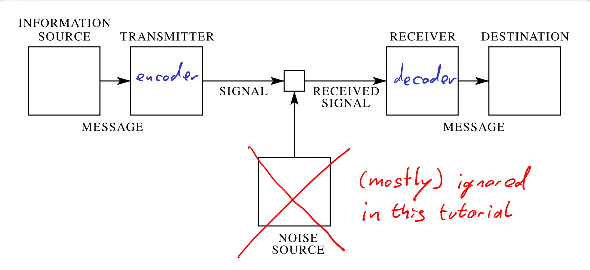


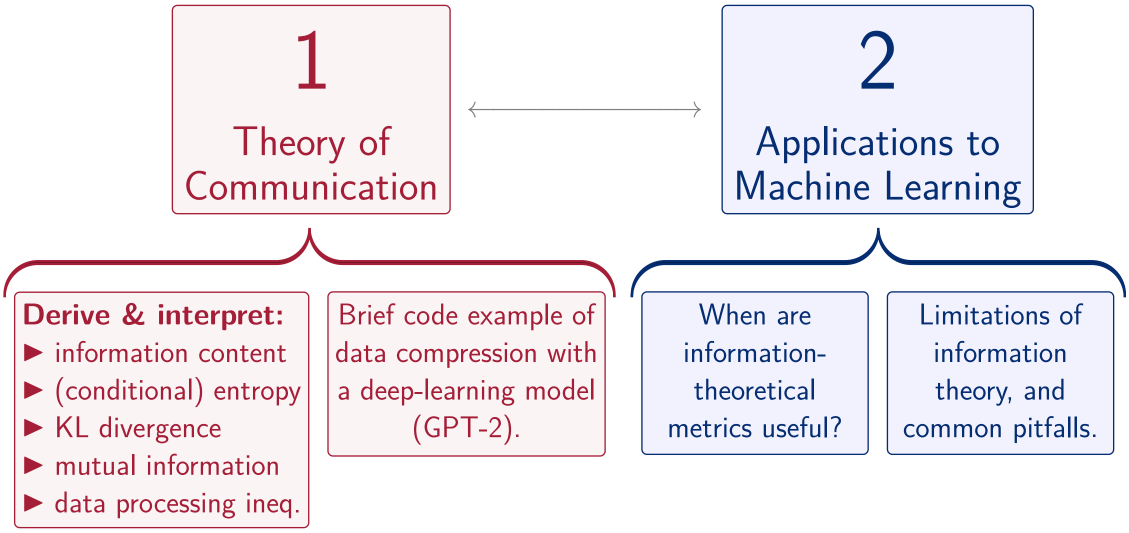
Fig. 1—Schematic diagram of a general communication system.

Def. “information content of a message”:

The *minimum number of bits* that you would have to transmit over a noise-free channel in order to communicate the message, *assuming an optimal encoder and decoder.*

- What does “optimal” mean?
- You don’t actually have to construct an optimal encoder & decoder to calculate this number.

Agenda



1

Theory of Communication

Derive & interpret:
- information content
- (conditional) entropy
- KL divergence
- mutual information
- data processing ineq.

Brief code example of data compression with a deep-learning model (GPT-2).

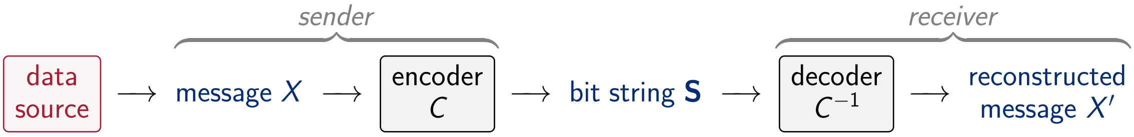
2

Applications to Machine Learning

When are information-theoretical metrics useful?

Limitations of information theory, and common pitfalls.

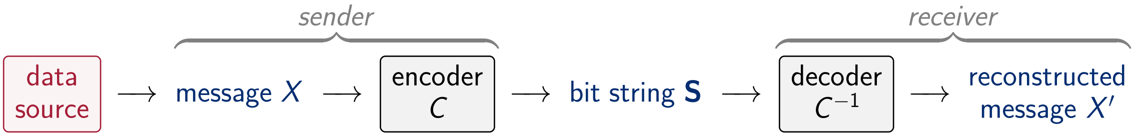
Data Compression: Precise Problem Setup



Assumptions:

- the bit string $\mathbf{S}$ is sent over a *noise free* channel (we won't cover *channel coding*);
- *lossless* compression: we require that $X' = X$; $(|S| = |C(x)| < \infty \ \forall x \in X$, but fct. $x \mapsto |C(x)|$ may be unbounded.)
- $\mathbf{S}$ may have a different length $|\mathbf{S}|$ for different messages: $\mathbf{S} \in \{0,1\}^* := \bigcup_{n=0}^{\infty} \{0,1\}^n$;
  - But: the encoder must *not* encode any information in the *length* of $\mathbf{S}$ alone (see next slide).
- Before the sender sees the message, sender and receiver can communicate arbitrarily much for free in order to agree on a code C : message space $\mathcal{X} \longrightarrow \{0,1\}^*$.
- **Goal:** find a valid code C that minimizes the expected bit rate $\mathbb{E}_{P_{\text{data source}}(X)}\big[|C(X)|\big]$.

What's a "Valid Code"? (Unique Decodability)



Recall:

- The bit string $\mathbf{S} = C(X) \in \{0,1\}^*$ can have different lengths for different messages X.
- We want to interpret its length $|\mathbf{S}|$ as the *amount of information* in the message X.
  - Seems to make sense: if the sender sends, e.g., a bit string of length 3 to the receiver, then they can't communicate more than 3 bits of information …
  - … unless the fact that $|\mathbf{S}| = 3$ already communicates some information. We want to forbid this.
- **Add additional requirement:** C must be *uniquely decodable:*
  - Sender may concatenate the encodings of *several* messages: $\mathbf{S} := C(X_1) \,\|\, C(X_2) \,\|\, C(X_3) \,\|\, \ldots$
  - Upon receiving $\mathbf{S}$, the receiver must still be able to detect where each part ends.

Source Coding Theorem

Theorem (Shannon, 1949): Consider a data source $P(X)$ over a discrete message space $\mathcal{X}$.

- **The bad news:** in expectation, lossless compression can't beat the entropy:

$$\forall \text{ uniquely decodable codes } C: \quad \mathbb{E}_P\big[|C(X)|\big] \geq \mathbb{E}_P\big[-\log_2 P(X)\big] =: H_P(X).$$

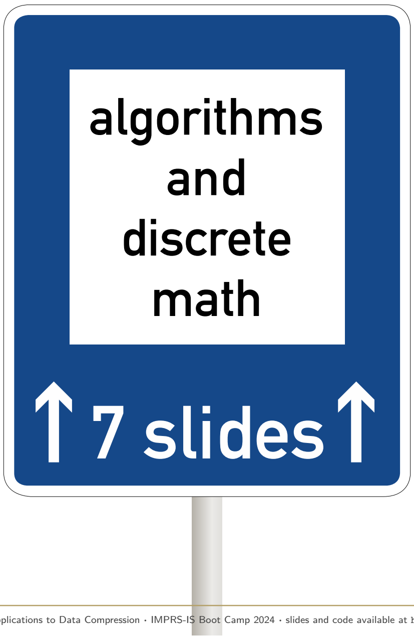
- **The good news:** but one can get quite close (and not just in expectation):

$$\exists \text{ uniquely decodable code } C:$$
$$\forall \text{ messages } x \in \mathcal{X}: \quad |C(x)| < -\log_2 P(X{=}x) + 1.$$
$$\big(\Longrightarrow \mathbb{E}_P\big[|C(X)|\big] < H_P(X) + 1\big)$$

Useful because it allows us to easily calculate (and differentiate through) the bit rate of an optimal code without having to explicitly construct it.

- Also, we can keep the total overhead < 1 bit even when encoding *several* messages.

$\Longrightarrow$ $-\log_2 P(X{=}x)$ is the contribution of message x to the bit rate of an optimal code when we *amortize* over many messages. It is called **"information content of x"**.



The Kraft-McMillan Theorem [Kraft, 1949; McMillan, 1956]

(a) $\forall$ uniquely decodable codes $C : \mathcal{X} \to \{0,1\}^*$ over some message space $\mathcal{X}$:

$$\text{(not an expectation, just a normal sum)} \longrightarrow \sum_{x\in\mathcal{X}} 2^{-|C(x)|} \leq 1 \qquad (\text{"Kraft inequality"}).$$

Interpretation: we have a finite budget of "shortness" for bit strings:

- Interpret $2^{-|C(x)|}$ as the "shortness" of bit string $C(x)$.
- The sum of all "shortnesses" must not exceed 1.

$\Longrightarrow$ If we shorten one bit string then we may have to make another bit string longer so that we don't exceed our "shortness budget".

(b) $\forall$ functions $\ell : \mathcal{X} \to \mathbb{N}$ that satisfy the Kraft inequality (i.e., $\sum_{x\in\mathcal{X}} 2^{-\ell(x)} \leq 1$):
$\exists$ uniquely decodable code C_ℓ with $|C_\ell(x)| = \ell(x) \;\; \forall x \in \mathcal{X}$.

Why is this theorem useful? $\Longrightarrow \min \mathbb{E}_P[\text{bit rate}] = \min\limits_{\substack{C\text{: uniq.}\\ \text{decodable}}} \mathbb{E}_P\big[|C(X)|\big] = \min\limits_{\substack{C\text{: satisfies}\\ \text{Kraft ineq.}}} \mathbb{E}_P\big[|C(X)|\big]$

Preparations for Proof of KM Theorem

Definition: For a code $C : \mathcal{X} \to \{0,1\}^*$, define

$$C^* : \mathcal{X}^* \to \{0,1\}^*, \quad C^*\big((x_1, x_2, \ldots, x_k)\big) := C(x_1) \parallel C(x_2) \parallel \ldots \parallel C(x_k).$$

(Thus: C is uniquely decodable $\iff$ C^* is injective)

Lemma:

- let: $\begin{cases} C \text{ be a uniquely decodable code over } \mathcal{X}; \\ n \in \mathbb{N}_0; \\ Y_n := \big\{\mathbf{x} \in \mathcal{X}^* \text{ with } |C^*(\mathbf{x})| = n\big\}. \end{cases}$
- then: $|Y_n| \leq 2^n$.

Proof:

$$\left.\begin{aligned} &C^* \text{ is injective} &&\Rightarrow |Y_n| = |C^*(Y_n)| \\ &C^*(Y_n) \subseteq \{0,1\}^n &&\Rightarrow |C^*(Y_n)| \leq |\{0,1\}^n| = 2^n \end{aligned}\right\} \Rightarrow |Y_n| \leq 2^n \qquad \square$$

Proof of Part (a) of KM Theorem

Lemma (reminder): $|Y_n| \leq 2^n$ where $Y_n := \big\{\mathbf{x} \in \mathcal{X}^* \text{ with } |C^*(\mathbf{x})| = n\big\}$, C uniq. dec.

Claim (reminder): C is uniquely decodable $\implies$ $\underbrace{\sum_{x \in \mathcal{X}} 2^{-|C(x)|}}_{=:r} \leq 1.$

Let $k \in \mathbb{N}$;

$$r^k = \Big(\sum_{x_1 \in \mathcal{X}} 2^{-|C(x_1)|}\Big)\Big(\sum_{x_2 \in \mathcal{X}} 2^{-|C(x_2)|}\Big) \cdots \Big(\sum_{x_k \in \mathcal{X}} 2^{-|C(x_k)|}\Big) = \sum_{\underline{x} \in \mathcal{X}^k} 2^{-\underbrace{\sum_{i=1}^{k} |C(x_i)|}_{=|C^*(\underline{x})|}}$$

(i) if $\mathcal{X}$ is finite:

$$\text{Let } \gamma := \max_{x \in \mathcal{X}} |C(x)| < \infty \Rightarrow \forall \underline{x} \in \mathcal{X}^k: |C^*(\underline{x})| \leq k\gamma \Rightarrow \mathcal{X}^k \subseteq \bigcup_{n=0}^{k\gamma} Y_n$$

$$\Rightarrow r^k \leq \sum_{n=0}^{k\gamma} \sum_{\underline{x} \in Y_n} 2^{-\underbrace{|C^*(\underline{x})|}_{=n}} = \sum_{n=0}^{k\gamma} \underbrace{|Y_n|}_{\leq 2^n} 2^{-n} \leq k\gamma + 1 \Rightarrow \forall k \in \mathbb{N}: \underbrace{\frac{r^k}{k}}_{\substack{\xrightarrow{k\to\infty} \infty \\ \text{unless } r \leq 1}} \leq \underbrace{\gamma + \frac{1}{k}}_{\xrightarrow{k\to\infty} \gamma < \infty}$$

(ii) if $\mathcal{X}$ is countably infinite:

$$\text{w.r.: } \mathcal{X} = \mathbb{N} \Rightarrow \sum_{x \in \mathcal{X}} 2^{-|C(x)|} \overset{\substack{\text{all terms} \geq 0 \\ \downarrow}}{=} \sum_{x=1}^{\infty} 2^{-|C(x)|} = \lim_{N\to\infty} \overbrace{\sum_{x=1}^{N} 2^{-|C(x)|}}^{\leq 1 \text{ by (i)}} \leq 1$$

Proof of Part (b) of KM Theorem

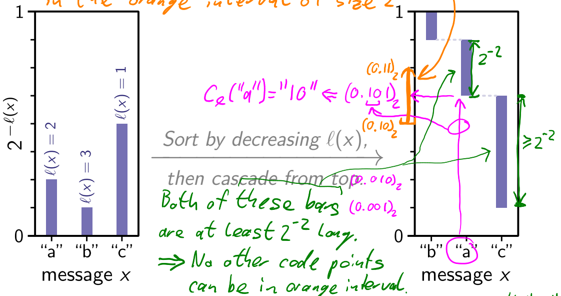
Note: computationally efficient variants of this idea: arithmetic coding and range coding

Claim (reminder): $\sum_{x \in \mathcal{X}} 2^{-\ell(x)} \leq 1 \implies \exists$ uniq. dec. code C_ℓ with $|C_\ell(x)| = \ell(x)\ \forall x \in \mathcal{X}$.

All binary numbers that begin with "0.10" lie in the orange interval of size 2^{-2}

Algorithm 1: Construction of C_ℓ.

```
Initialize $\xi \leftarrow 1$;
for $x \in \mathcal{X}$ in order of nonincreasing $\ell(x)$ do
    Update $\xi \leftarrow \xi - 2^{-\ell(x)}$;
    Write $\xi \in [0, 1)$ in binary: $\xi = (0.???\ldots)_2$;
    Set $C_\ell(x)$ to the first $\ell(x)$ bits after the "0."
        (pad with trailing zeros if necessary);
```



$C_\ell(\text{"a"}) = \text{"10"} \Leftarrow (0.\underline{10}1)_2$

Both of these bars are at least 2^{-2} long.

$\Rightarrow$ No other code points can be in orange interval.

$\Rightarrow$ No other $C_\ell(x')$, $x' \neq$ "a" can begin with "10".

Claim: the resulting code C_ℓ is uniquely decodable.

- We even show: C_ℓ is *prefix free:* $\forall x \in \mathcal{X}$: $C_\ell(x)$ is not the beginning of any $C_\ell(x')$, $x' \neq x$.
  ↳ trivially uniq. dec.
- Formalization of this proof: see solutions to Problem 2.1 on this problem set:
  `https://robamler.github.io/teaching/compress23/problem-set-02-solutions.zip`

Example: Sum of Two Fair 3-Sided Dice

Note: no encoding $C_\ell(x)$ is a prefix of a different encoding $C_\ell(x')$ with $x' \neq x$ $\Rightarrow C_\ell$ is uniquely decodable

x	possible throws	$P(X=x)$	$\ell(x)$	$C_\ell(x)$
2	⚀⚀	1/9	3	111
3	⚀⚁, ⚁⚀	2/9	2	10
4	⚀⚂, ⚁⚁, ⚂⚀	1/3	2	01
5	⚁⚂, ⚂⚁	2/9	2	00
6	⚂⚂	1/9	3	110

step	
	$\xi \leftarrow 1 = (1.000)_2$
①	$\xi \leftarrow 1 - 2^{-3} = (1.000)_2 - (0.001)_2 = (0.\underline{111})_2$
③	$\xi \leftarrow (0.110)_2 - 2^{-2} = (0.11)_2 - (0.01)_2 = (0.\underline{10})_2$
④	$\xi \leftarrow (0.10)_2 - (0.01)_2 = (0.\underline{01})_2$
⑤	$\xi \leftarrow (0,01)_2 - (0.01)_2 = (0.\underline{00})_2$
②	$\xi \leftarrow (0.111)_2 - 2^{3} = (0.111)_2 - (0.001)_2 = (0.\underline{110})_2$

- Check if ℓ satisfies Kraft inequality: $\sum_{x\in\mathcal{X}} 2^{-\ell(x)} = 2 \times 2^{-3} + 3 \times 2^{-2} = 1 \leqslant 1$ ✓
- **Question:** how should we choose $\ell : \mathcal{X} \to \mathbb{N}$ for a given model P of the data source?
  - **typical goal:** minimize the expected bit rate $\mathbb{E}_P[\ell(X)]$ (with the constraint $\sum_{x\in\mathcal{X}} 2^{-\ell(x)} \leq 1$).
  - **optimally:** by *Huffman coding* (comp. cost $\propto |\mathcal{X}| \log |\mathcal{X}|$, i.e., exponential in message length).
  - **near optimally:** via *information content*; → bounds on optimal $\mathbb{E}_P[\ell(X)]$ (next slide).

(problem: no closed-form expression for optimal $\ell(x)$ & not differentiable)

Optimal Choice of Target Length $\ell : \mathcal{X} \to \mathbb{N}$

- **Constrained optimization problem:**
  - Minimize $\mathbb{E}_P[\ell(X)] = \sum_{x\in\mathcal{X}} P(X=x)\,\ell(x)$ over ℓ
  - with the constraints: (i) $\sum_{x\in\mathcal{X}} 2^{-\ell(x)} \leq 1$

    (ii) $\ell(x) \in \mathbb{N} \;\; \forall x \in \mathcal{X}$
- **Idea:** relax constraint: (ii') $\ell(x) \in \mathbb{R}_{>0} \;\; \forall x \in \mathcal{X}$

  $\Rightarrow$ Minimization runs over more functions ℓ.

  $\Rightarrow$ *lower bound:* $\inf_{\text{(i),(ii')}} \mathbb{E}_P[\ell(X)] \leq \inf_{\text{(i),(ii)}} \mathbb{E}_P[\ell(X)]$
- **Observation:** solution satisfies: (i') $\sum_{x\in\mathcal{X}} 2^{-\ell(x)} = 1$
  - Enforce via Lagrange multiplier $\lambda \in \mathbb{R}$:

    find stationary point (w.r.t. both ℓ and λ) of $\mathcal{L}(\ell,\lambda) := \underline{\sum_{x\in\mathcal{X}} P(X=x)}\,\ell(x) + \lambda\big(\underline{\sum_{x\in\mathcal{X}} 2}^{-\ell(x)} - 1\big)$.

$$0 = \frac{\partial \mathcal{L}}{\partial \lambda} = \sum_{x\in\mathcal{X}} 2^{-\ell(x)} - 1 \quad \text{(constraint (i'))}$$

$$2^{-\ell(x)} = \exp\left[\ln\left(2^{-\ell(x)}\right)\right] = \exp\left[-\ell(x)\ln 2\right]$$

$$\forall x:\; 0 = \frac{\partial \mathcal{L}}{\partial \ell(x)} = \frac{\partial}{\partial \ell(x)} \sum_{x'\in\mathcal{X}} \Big[\underline{P(X=x')\,\ell(x')} + \underline{+\lambda \exp[-\ell(x)\ln 2]}\Big]$$

$$= P(X=x) - \lambda(\ln 2)\,2^{-\ell(x)}$$

Solve for $\ell(x)$:

$$\ell(x) = -\log_2 \frac{P(X=x)}{\underbrace{\lambda \ln 2}_{=1}} = -\log_2 P(X=x)$$

Insert constraint (i'):

$$1 = \sum_{x\in\mathcal{X}} 2^{-\ell(x)} = \frac{1}{\lambda \ln 2} \overbrace{\sum_{x\in\mathcal{X}} P(X=x)}^{=1}$$

Proof of Source Coding Theorem

- Solution of the relaxed optimization problem: $\ell(x) = \underbrace{-\log_2 P(X=x)}_{\text{“information content”}} \in \mathbb{R}_{\geq 0}$.
- Let's now constrain $\ell(x)$ again to integer values $\forall x \in \mathcal{X}$.

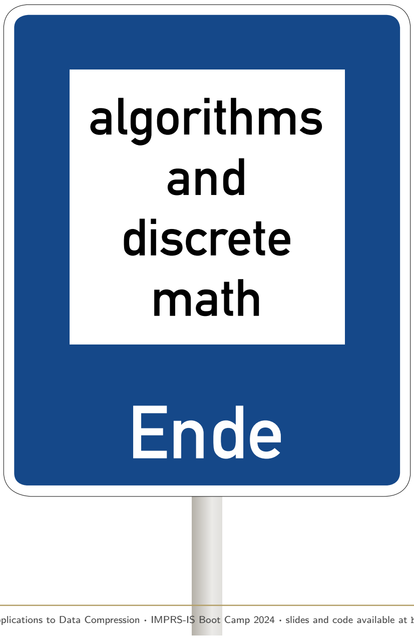
  $\Longrightarrow$ **lower bound** on expected bit rate (“the bad news”):

$$\mathbb{E}_P\big[|C(X)|\big] \geq \underbrace{\mathbb{E}_P\big[-\log_2 P(X=x)\big]}_{H_P(X)} \qquad \forall \text{ uniquely decodable } C.$$

- **Upper bound** on the *optimal* expected bit rate (“the good news”):
  - *Shannon Code:* set $\ell(x) := \lceil -\log_2 P(X=x) \rceil \in \mathbb{N}$.
  - Satisfies Kraft inequality: $\sum_{x\in\mathcal{X}} 2^{-\lceil -\log_2 P(X=x)\rceil} \leq \sum_{x\in\mathcal{X}} 2^{\log_2 P(X=x)} = 1$.

    $\Longrightarrow \exists$ uniquely decodable code C_ℓ with:

$$|C_\ell(x)| = \ell(x) < -\log_2 P(X=x) + 1 \qquad \forall x \in \mathcal{X}.$$



Quantifying Uncertainty in Bits (for Discrete Data)

- **Information content:** $-\log_2 P(X{=}x)$**:** The (amortized) bit rate for encoding the given message x with a code that is optimal (in expectation) for the data source P.
- **Entropy:** $H_P(X) = \mathbb{E}_P\big[-\log_2 P(X)\big] \equiv H\big[P(X)\big] \equiv H[P]$**:** The *expected* bit rate for encoding a (random) message from data source P with a code that is optimal for P.
  $=$ How many bits does receiver need (in expectation) to reconstruct X?
  $=$ How many bits does receiver need (in expectation) to resolve any *uncertainty* about X?
- **Cross entropy:** $H[P,Q] = \mathbb{E}_P\big[-\log_2 Q(X)\big] \geq H[P]$**:**
  The expected bit rate when encoding a message from data source P with a code that is optimal for a model Q of the data source ($\Longrightarrow$ practically achievable expected bit rate).
  $\rightarrow$ We'd want to minimize this over the model Q. $\rightarrow$ Maximum likelihood estimation.
- **Kullback-Leibler divergence:** $D_{\mathrm{KL}}(P\,\|\,Q) = H[P,Q] - H[P] = \mathbb{E}_P\Big[-\log_2 \frac{Q(X)}{P(X)}\Big] \geq 0$**:**
  Overhead (in expected bit rate) due to a mismatch between the true data source P and its model Q (also called *"relative entropy"*).

D_{KL} can be ∞ (if $\exists x_0$ where $P(X{=}x_0) > 0$ but $Q(X{=}x_0) = 0$).
$\Rightarrow$ Interpretation: optimal code for model Q could not encode x_0.

Agenda

1
Theory of Communication

⟷

2
Applications to Machine Learning

Derive & interpret:
- information content
- (conditional) entropy
- KL divergence
- mutual information
- data processing ineq.

Brief code example of data compression with a deep-learning model (GPT-2).

When are information-theoretical metrics useful?

Limitations of information theory, and common pitfalls.

Example 1: Text Compression With GPT-2

Autoregressive language model:

- Message **x** is a sequence of tokens: $\mathbf{x} = (x_1, x_2, \ldots, x_n)$.
- $P(\mathbf{X}) = P(X_1)\,P(X_2 \,|\, X_1)\,P(X_3 \,|\, X_1, X_2)\,P(X_4 \,|\, X_1, X_2, X_3)\ \ldots\ P(X_n \,|\, X_1, X_2, \ldots X_{n-1})$.

(We take expectation over our <u>model</u> P here rather than over the true (cultural) data gen. process; so there's an <u>additional</u> overhead due to model mismatch that we don't discuss here.)

Compression strategy:

1. Encode x_1 with an optimal code for $P(X_1)$. $\rightarrow \mathbb{E}_{\textcircled{P}}[\#\text{bits}] < H\big[P(X_1)\big] + 1$
2. Encode x_2 with an optimal code for $P(X_2 \,|\, X_1{=}x_1)$. $\rightarrow \mathbb{E}_{\textcircled{P}}[\#\text{bits}] < H\big[P(X_2 \,|\, X_1{=}x_1)\big] + 1$
3. And so forth ... (decoder operates in same order as encoder)

Technicalities:

`https://bamler-lab.github.io/bootcamp24` → *Colab notebook*

- Up to 1 bit of overhead *per token*? → Use a *stream code:* amortizes over tokens.
- The model expects that $x_1 = \langle$*beginning of sequence*$\rangle$. → Redundant, don't encode.
- How does the *decoder* know when to stop? → Use an $\langle$*end of sequence*$\rangle$ token.

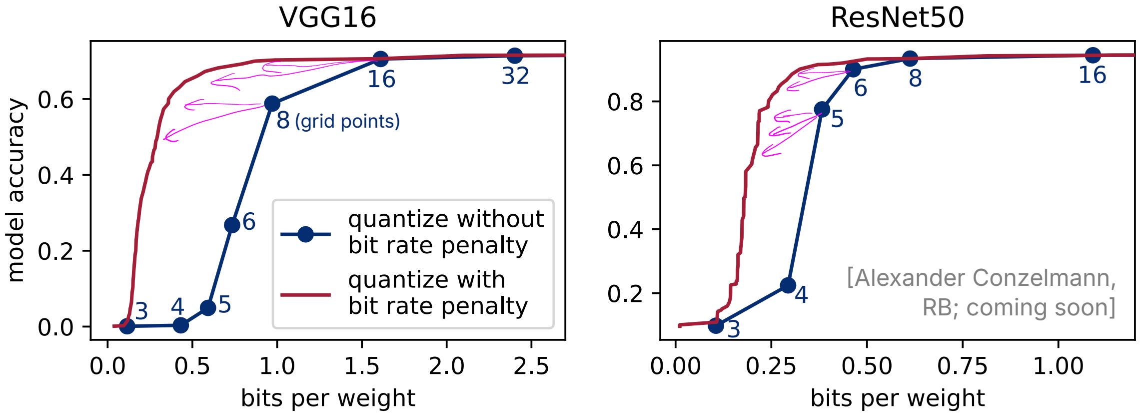
Takeaways From Our Code Example

- Near-optimal compression performance is achievable *in practice*.

  $\Longrightarrow$ Information content accurately estimates #bits needed *in practice* (even if it's fractional).
- Data compression is intimately tied to *probabilistic generative modeling*.
  - *"Don't transmit what you can predict."* $\Longrightarrow$ generative modeling
  - But still allow communicating things we wouldn't have predicted. $\Longrightarrow$ probabilistic modeling
- Decoding ≈ generation (= sampling from a probabilistic generative model P):
  - To *sample* a token x_i, one injects *randomness* into $P(X_i \,|\, \mathbf{X}_{1:i-1}{=}\mathbf{x}_{1:i-1})$.
  - To *decode* a token x_i, one injects *compressed bits* into (a code for) $P(X_i \,|\, \mathbf{X}_{1:i-1}{=}\mathbf{x}_{1:i-1})$.
  - Decoding from a *random* bit string would be exactly equivalent to sampling from P.
- Data compression is highly sensitive to tiny model changes (e.g., inconsistent rounding).
  - Compression codes C are "very non-continuous" (because they *remove redundancies* by design).

  $\Longrightarrow$ True data compression usually makes it *harder* to process information.

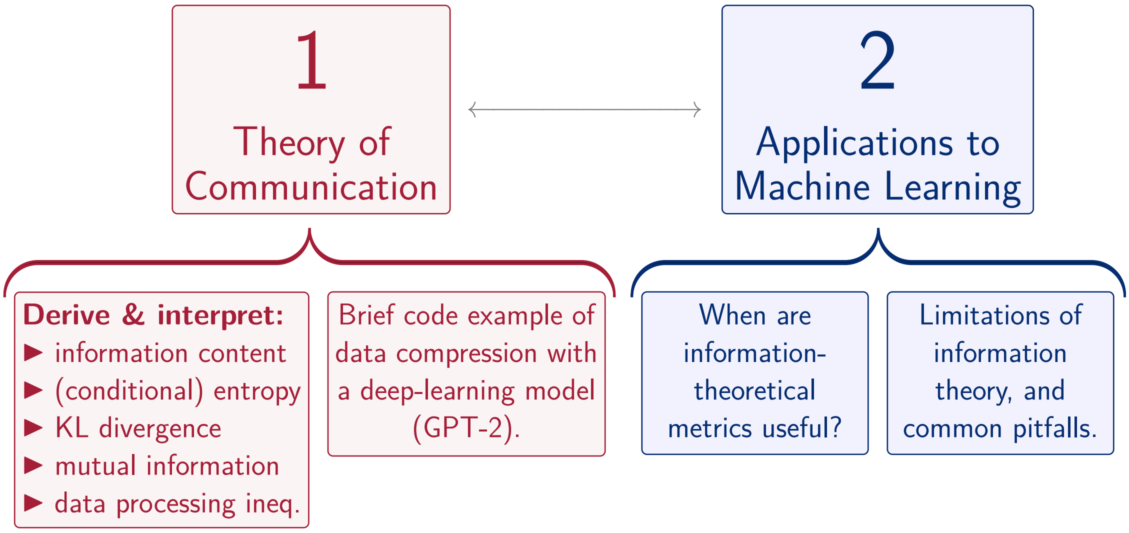
Example 2: Compression ~~With~~ *of* Neural Networks



[Alexander Conzelmann, RB; coming soon]

- **Method:** quantize network weights (≈ round to a discrete grid), then compress losslessly.
- **Observation:** information content remains meaningful *even in the regime* $\ll 1$ bit.

Agenda



1

Theory of Communication

2

Applications to Machine Learning

Derive & interpret:
- information content
- (conditional) entropy
- KL divergence
- mutual information
- data processing ineq.

Brief code example of data compression with a deep-learning model (GPT-2).

When are information-theoretical metrics useful?

Limitations of information theory, and common pitfalls.

Joint, Marginal, and Conditional Entropy

Consider a data source $P(X, Y)$ that generates pairs $(x, y) \sim P$:

$$P(X, Y) = P(X)\, P(Y \,|\, X) = P(Y)\, P(X \,|\, Y).$$

- **Joint information content**, i.e., information content of the entire message (x, y): $-\log_2 P(X{=}x, Y{=}y) = -\log_2 P(X{=}x) - \log_2 P(Y{=}y \,|\, X{=}x)$.
- **Joint entropy:**
$$\begin{aligned} H_P\big((X, Y)\big) &= \mathbb{E}_{P(X,Y)}[-\log_2 P(X, Y)] = \mathbb{E}_{P(X)\,P(Y|X)}\big[-\log_2 P(X) - \log_2 P(Y|X)\big] \\ &= \underbrace{\mathbb{E}_{P(X)}[-\log_2 P(X)]}_{\text{(marginal) entropy } H_P(X)} + \underbrace{\mathbb{E}_{x\sim P(X)}\Big[\underbrace{\mathbb{E}_{P(Y|X=x)}[-\log_2 P(Y \,|\, X{=}x)]}_{=:\, H_P(Y \,|\, X=x) \,=\, \textbf{entropy of the conditional distribution } P(Y \,|\, X=x)}\Big]}_{=:\, \textbf{conditional entropy } H_P(Y \,|\, X)}; \end{aligned}$$
- $$\boxed{H_P\big((X, Y)\big) = H_P(X) + H_P(Y \,|\, X) = H_P(Y) + H_P(X \,|\, Y)}$$

Mutual Information

Reminder: $H_P(Y \,|\, X) := \mathbb{E}_P\big[-\log_2 P(Y \,|\, X)\big] = \mathbb{E}_{x\sim P(X)}\Big[\underbrace{\mathbb{E}_{P(Y|X=x)}[-\log_2 P(Y \,|\, X{=}x)]}_{=\, H_P(Y \,|\, X=x) \,=\, \text{entropy of the conditional distribution } P(Y \,|\, X=x)}\Big]$;

$H_P\big((X, Y)\big) = H_P(X) + H_P(Y \,|\, X)$.

Let's encode a given message (x, y):

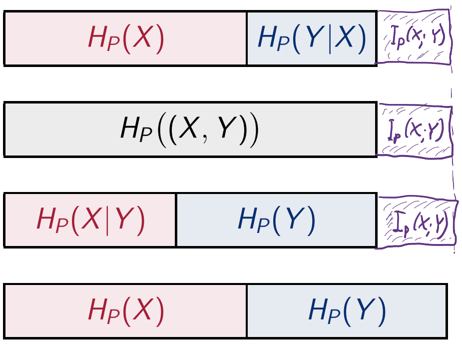
(a) encode x with optimal code for $P(X)$; then enocde y with optimal code for $P(Y \,|\, X{=}x)$;

(b) encode (x, y) using an optimal code for the data source $P(X, Y)$;

(c) encode x with optimal code for $P(X \,|\, Y{=}y)$; then enocde y with optimal code for $P(Y)$.

(d) encode x with optimal code for $P(X)$; then enocde y with optimal code for $P(Y)$;

Expected bit rate:



Interpretations of the Mutual Information $I_P(X;Y)$

The following expressions for $I_P(X;Y)$ are equivalent:

① $I_P(X;Y) = H_P(X) + H_P(Y) - H_P((X,Y))$
$\qquad = D_{\mathrm{KL}}\big(P(X,Y)\,\|\,P(X)\,P(Y)\big) \geq 0$

Interpretation: how much would ignoring correlations between X, Y hurt expected compression performance?

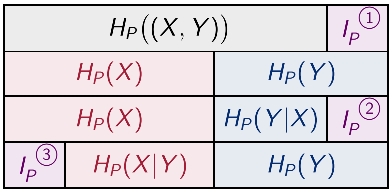
↖ (i.e., how many bits can we save if we take correlations into account?)

② $I_P(X;Y) = H_P(Y) - H_P(Y\,|\,X)$

Interpretation: how many bits of information does knowledge of X tell us about Y (in expectation)?
(reduction of uncertainty, *"expected information gain"*)

③ $I_P(X;Y) = H_P(X) - H_P(X\,|\,Y)$

Interpretation: how many bits of information does knowledge of Y tell us about X (in expectation)?



Note: "in expectation" is an important qualifier here. Conditioning on a *specific* x can *increase* the entropy of Y:

$H_P(Y\,|\,X) \leq H_P(Y)$ (always),

but:

$H_P(Y\,|\,X{=}x) > H_P(Y)$ is possible for some (atypical) x.

Continuous Data (Pedestrian Approach)

Recall: optimal lossless code C_{opt} for a data source P: $\boxed{H_P(X) \leq \mathbb{E}_P\big[|C_{\mathrm{opt}}(X)|\big] < H_P(X) + 1}$

- Lossless compression is only possible on a *discrete* (i.e., countable) message space $\mathcal{X}$.
  (Because $\mathcal{X} \xrightarrow{\text{lossless code } C \text{ (injective)}} \{0,1\}^* \xrightarrow{\text{injective}} \mathbb{N}$.)

Simple *lossy* compression of a message $X \in \mathbb{R}^n$: (an "act of desperation" — M.P.)

- Require that reconstruction X' satisfies $|X'_i - X_i| < \frac{\delta}{2}\ \forall i \in \{1,\ldots,n\}$ for some $\delta > 0$.
- Let $\hat{X} := \delta \times \mathrm{round}\big(\frac{1}{\delta}X\big)$. $\quad\Longrightarrow |\hat{X}_i - X_i| \leq \frac{\delta}{2}\ \ \forall i$.
- Compress $\hat{X} \in \delta\mathbb{Z}^n$ losslessly using induced model $P(\hat{X})$. $\quad\Longrightarrow$ Reconstruction $X' = \hat{X}$.
- $P(\hat{X}{=}\hat{x}) = P\Big(X \in \bigtimes_{i=1}^{n}\big[\hat{x}_i - \frac{\delta}{2}, \hat{x}_i + \frac{\delta}{2}\big)\Big) = \int_{\times_{i=1}^{n}[\hat{x}_i-\frac{\delta}{2},\hat{x}_i+\frac{\delta}{2})} p(x)\,\mathrm{d}^n x \approx \delta^n p(\hat{x}) + o(\delta^n)$
- $H_P(\hat{X}) \approx -\sum_{\hat{x}\in\delta\mathbb{Z}^n} \delta^n p(\hat{x}) \log_2\big(\delta^n p(\hat{x})\big)$

$$\approx -\int p(x)\,\big(\log_2 p(x) + n\log_2\delta\big)\,\mathrm{d}^n x = \underbrace{\mathbb{E}_P[-\log_2 p(X)]}_{\text{"differential entropy" } h_P(X)} + \underbrace{n\log_2(1/\delta)}_{\xrightarrow{\delta\to 0}\ \infty}$$

⇒ Entropy of a continuous random variable is "defined up to an infinite constant offset.

How Does Discretization Relate to IMPRS-IS?

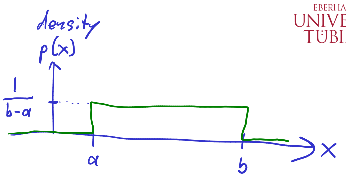
Physics in the 19th century:

- **Electrodynamics:** unified theory of electric+magnetic forces (Lorentz, Maxwell, ∼1860)
  ⟶ understanding of light ⟶ radio communication (Marconi, ∼1895)
- **Thermodynamics:** temperature, heat, entropy, *steam engine* (Carnot process)

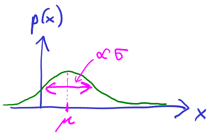
Problem: these two theories are incompatible when trying to explain the spectrum of the sun.

- **"Classical" theory:** it should radiate ∞ energy ("ultraviolet catastrophe").
  - **Electrodynamics:** energy E_f of electromagnetic field at frequency f is a *continuous* quantity $\forall f$.
  - **Thermodynamics:** thus, in thermodynamic equilibrium, $\mathbb{E}[E_f] = \frac{1}{2}k_{\mathrm{B}}T\ \ \forall f$.
- **Observation:** OK for low frequencies ($\exists < \infty$), wrong for high frequencies ($\exists\ \infty$).
- **Max Planck, 1900:** discrepancies can be resolved if we assume that $\boxed{E_f \in hf\mathbb{Z}\quad \forall f}$.
  - **Quantum mechanics** becomes relevant on energy scales $E \lesssim hf$, with the *Planck constant* $h \approx 6.626\times10^{-34}\,\frac{\mathrm{J}}{\mathrm{Hz}}$; foundation of modern chemistry, semiconductor industry, ...

Examples of Differential Entropies



- **Uniform distribution:** $P(X) = \mathcal{U}(\mathcal{X})$
  - **Density:** $p(x) = \frac{1}{\mathrm{Vol}(\mathcal{X})} \quad \forall x \in \mathcal{X}$
  - **Differential entropy:** $h_P(X) = \mathbb{E}_P\big[-\log_2 p(X)\big] = \mathbb{E}_P\left[-\log_2 \frac{1}{\mathrm{Vol}(\mathcal{X})}\right] = \log_2\big(\mathrm{Vol}(\mathcal{X})\big)$
  - **Note:** if $\mathrm{Vol}(\mathcal{X}) < 1$ then $h_P(X) < 0$. (c.f., acidic solution with pH < 0: not special)
    $\rightarrow$ Nothing to see here, h_P is only meaningful up to an infinite additive constant.



- **Normal distribution:** $P(X) = \mathcal{N}(\mu, \Sigma)$ (with $X, \mu \in \mathbb{R}^n$ and $\Sigma \in \mathbb{R}^{n\times n}$)
  in 1d: $\Sigma = \sigma^2$
  - **Density:** $p(x) = \mathcal{N}(x; \mu, \Sigma) = \frac{1}{\sqrt{\det(2\pi\,\Sigma)}} \exp\left[-\frac{1}{2}(x-\mu)^\top \Sigma^{-1} (x-\mu)\right]$
  - **Differential entropy:** $h_P(X) = \mathbb{E}_P\big[-\log_2 p(X)\big] = \frac{1}{2}\log_2(\det\Sigma) + \underbrace{\frac{n}{2}\log_2(2\pi e)}_{\text{const.}}$
    (plausibility check: large variance ⇔ large entropy ✓)

KL-Divergence Between Continuous Distributions

- **Differential entropy** (reminder): $h_P(X) = \mathbb{E}_P[-\log_2 p(X)]$
  $\rightarrow$ Relation to entropy of discretization $\hat{X}$: $H_P(\hat{X}) \approx h_P(X) + n\log_2(1/\delta) \xrightarrow{\delta\to 0} \infty$
- **Differential cross entropy** (less common): $h\big[P(X), Q(X)\big] = \mathbb{E}_P[-\log_2 q(X)]$
  $\rightarrow$ Relation to discretization: $H\big[P(\hat{X}), Q(\hat{X})\big] \approx h\big[P(X), Q(X)\big] + n\log_2(1/\delta) \xrightarrow{\delta\to 0} \infty$
- **Kullback-Leibler divergence** between discretized distributions $P(\hat{X})$ and $Q(\hat{X})$:

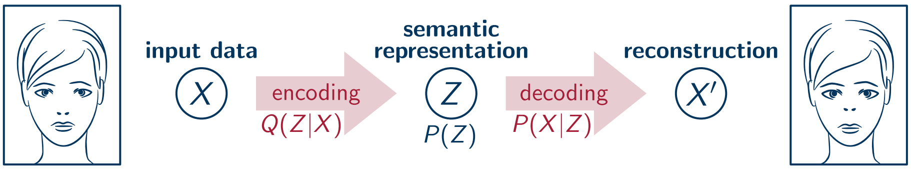
$$\begin{aligned}
D_{\mathrm{KL}}\big(P(\hat{X}) \,\|\, Q(\hat{X})\big) &= H\big[P(\hat{X}), Q(\hat{X})\big] - H_P(\hat{X}) \\
&\approx h\big[P(X), Q(X)\big] + n\log_2(1/\delta) - \big(h_P(X) + n\log_2(1/\delta)\big) \\
&= \mathbb{E}_P\left[-\log_2 \frac{q(X)}{p(X)}\right] =: D_{\mathrm{KL}}\big(P(X) \,\|\, Q(X)\big) \overset{\text{(possibly)}}{<} \infty
\end{aligned}$$

$\Longrightarrow$ **Interpretation:** $D_{\mathrm{KL}}(P \,\|\, Q) =$ modeling overhead, *in the limit of infinitely fine quantization.*

- **Generalization (density-free):** $D_{\mathrm{KL}}(P \,\|\, Q) = -\int \log_2\left(\frac{\mathrm{d}Q}{\mathrm{d}P}\right)\mathrm{d}P$

(Variational) Information Bottleneck

- **Example:** β-variational autoencoder (similar for supervised models (Alemi et al., ICLR 2017))

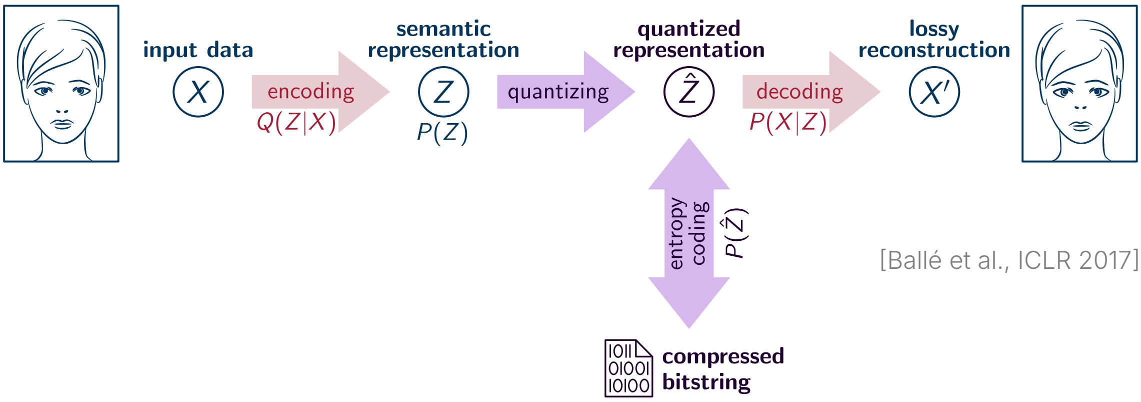


- **Loss function:** $\mathbb{E}_{x\sim\text{data}}\Big[\mathbb{E}_{Q(Z|X=x)}\big[-\log P(X{=}x \,|\, Z)\big] + \beta D_{\mathrm{KL}}\big(Q(Z\,|\,X{=}x) \,\|\, P(Z)\big)\Big]$
- $D_{\mathrm{KL}}(\ldots\|\ldots) =$ $\boxed{\begin{array}{c}\text{information in } z \sim Q(Z\,|\,X{=}x) \\ \text{for someone who doesn't know } x \\ \text{(i.e., they only know } P(Z))\end{array}} - \boxed{\begin{array}{c}\text{information in } z \sim Q(Z\,|\,X{=}x) \\ \text{for someone who knows } x \\ \text{(i.e., they know } Q(Z\,|\,X{=}x))\end{array}}$

$\Longrightarrow \begin{cases}\text{Capture as much (}x\text{-independent) information about } z \text{ in the prior } P(Z) \text{ as possible.} \\ \text{Encode as little (unnecessary) information in } Q(Z\,|\,X{=}x) \text{ as possible.}\end{cases}$

Remark: Data Compression With VAEs

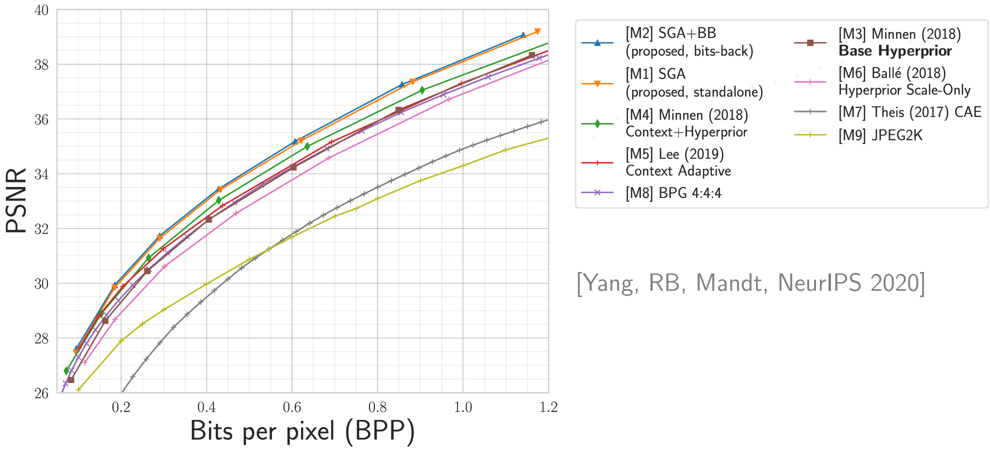
- **So far, no compression:** Z still takes up lots of memory (even if its inf. content is low).
- Real compression has to actually reduce Z to its information content: *entropy coding*



[Ballé et al., ICLR 2017]

Rate/Distortion Trade-off

- Tuning β allows us to trade off *bit rate* against *distortion*.



[Yang, RB, Mandt, NeurIPS 2020]

originals
(uncompressed)

BPG 4:4:4
left: 0.143 bit/pixel
right: 0.14 bit/pixel

VAE-based
left: 0.142 bit/pixel
right: 0.13 bit/pixel
[Yang, RB, Mandt, NeurIPS 2020]

JPEG
left: 0.142 bit/pixel
right: 0.14 bit/pixel

Agenda

1

Theory of Communication

Derive & interpret:
- information content
- (conditional) entropy
- KL divergence
- mutual information
- data processing ineq.

Brief code example of data compression with a deep-learning model (GPT-2).

2

Applications to Machine Learning

When are information-theoretical metrics useful?

Limitations of information theory, and common pitfalls.

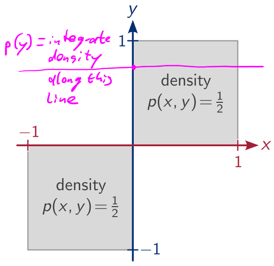
Mutual Information for Continuous Random Vars

$$I_P(X;Y) = D_{\mathrm{KL}}\big(P(X,Y) \,\big\|\, P(X)\,P(Y)\big) = \mathbb{E}_P\left[-\log_2 \frac{p(X)\,p(Y)}{p(X,Y)}\right]$$ (if densities p exist)

(we'll discuss non-injective fcts. later)

- **Exercise:** let $X' = f(X)$, $Y' = g(Y)$, where f and g are differentiable *injective* functions. Convince yourself that I_P is independent of representation, i.e., $I_P(X';Y') = I_P(X;Y)$.

Example 1a:

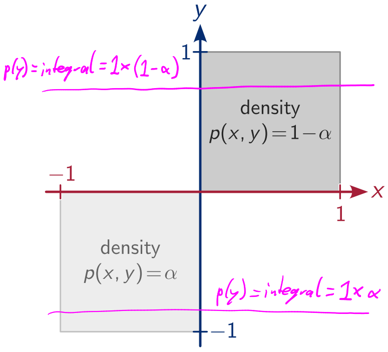


- $P(X) = P(Y) = \mathcal{U}([-1,1)) \implies h_P(X) = h_P(Y) = \log_2(2) = 1.$
- $P(Y\,|\,X) = \begin{cases} \mathcal{U}([-1,0)) & \text{if } X < 0; \\ \mathcal{U}([0,1)) & \text{if } X \geq 0. \end{cases}$

  $\implies h_P(Y\,|\,X) = \mathbb{E}_{x\sim P(X)}\big[h_P(Y\,|\,X{=}x)\big] = \log_2(1) = 0.$
- **Mutual information:** $I_P(X;Y) = h_P(Y) - h_P(Y\,|\,X) = 1 - 0 = 1$ bit.
- **Interpretation:** observing X tells us (only) the sign of Y. $\implies$ 1 bit of information.

Mutual Information for Continuous Random Vars

Example 1b: non-uniform $P(X)$.



- $p(y) = \begin{cases} \alpha & \text{if } y \in [-1,0); \\ 1-\alpha & \text{if } y \in [0,1). \end{cases}$ (for $\alpha \in [0,1]$)

  $$\Rightarrow h_P(Y) = -\int_{-1}^{1} p(y)\log_2 p(y)\,\mathrm{d}y$$
  $$= -\alpha\log_2(\alpha) - (1-\alpha)\log_2(1-\alpha) =: H_2(\alpha)$$
- $P(Y\,|\,X) = \begin{cases} \mathcal{U}([-1,0)) & \text{if } X < 0; \\ \mathcal{U}([0,1)) & \text{if } X \geq 0. \end{cases}$ (as before)

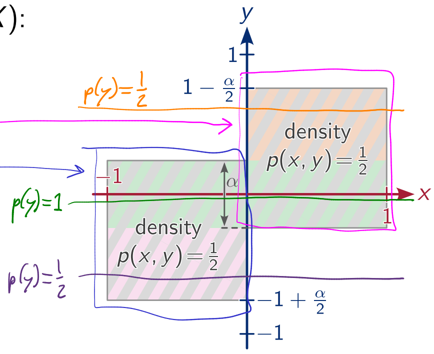
  $\Rightarrow h_P(Y\,|\,X) = 0$ (as before)
- **Mutual information:** $I_P(X;Y) = h_P(Y) - h_P(Y\,|\,X) = H_2(\alpha) - 0 = H_2(\alpha) \leq 1$ bit.
- **Interpretation:** observing X still tells us sign(Y) with certainty, but sign(Y) now carries less than one bit of information (in expectation) if $\alpha \neq \frac{1}{2}$.

Mutual Information for Continuous Random Vars

Example 1c: back to uniform $P(X)$, but different $P(Y\,|\,X)$:

- $P(X) = \mathcal{U}([-1, 1))$;

$$P(Y\,|\,X) = \begin{cases} \boxed{\mathcal{U}\big(\big[-\tfrac{\alpha}{2}, 1-\tfrac{\alpha}{2}\big)\big) \quad \text{if } X \geq 0;} \\ \boxed{\mathcal{U}\big(\big[-1+\tfrac{\alpha}{2}, \tfrac{\alpha}{2}\big)\big) \quad \text{if } X < 0.} \end{cases}$$



Method 1: $I_P(X;Y) = h_P(Y) - h_P(Y\,|\,X)$

- $h_P(Y\,|\,X) = 0$ as before.
- $p(y) = \begin{cases} \frac{1}{2} & \text{if } y \in \big[\frac{\alpha}{2}, 1-\frac{\alpha}{2}\big) \cup \big[-1+\frac{\alpha}{2}, -\frac{\alpha}{2}\big); \\ 1 & \text{if } y \in \big[-\frac{\alpha}{2}, \frac{\alpha}{2}\big). \end{cases}$

$$\Longrightarrow h_P(Y) = -\int_{\frac{\alpha}{2}}^{1-\frac{\alpha}{2}} \tfrac{1}{2}\log_2\!\big(\tfrac{1}{2}\big)\,\mathrm{d}y \;-\int_{-1+\frac{\alpha}{2}}^{-\frac{\alpha}{2}} \tfrac{1}{2}\log_2\!\big(\tfrac{1}{2}\big)\,\mathrm{d}y \;-\int_{-\frac{\alpha}{2}}^{\frac{\alpha}{2}} 1\log_2(1)\,\mathrm{d}y = 1-\alpha.$$

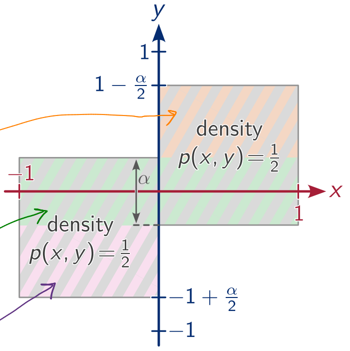
- **Interpretation:** sign(Y) has one bit of entropy again, but knowing X no longer tells us sign(Y) with certainty, it only improves our odds of predicting it.

Mutual Information for Continuous Random Vars

Example 1c: back to uniform $P(X)$, but different $P(Y\,|\,X)$:

- $P(X) = \mathcal{U}([-1, 1))$;

$$P(Y\,|\,X) = \begin{cases} \mathcal{U}\big(\big[-\tfrac{\alpha}{2}, 1-\tfrac{\alpha}{2}\big)\big) & \text{if } X \geq 0; \\ \mathcal{U}\big(\big[-1+\tfrac{\alpha}{2}, \tfrac{\alpha}{2}\big)\big) & \text{if } X < 0. \end{cases}$$



Method 2: $I_P(X;Y) = h_P(X) - h_P(X\,|\,Y)$

- $P(X\,|\,Y) = \begin{cases} \mathcal{U}([0,1)) & \text{if } Y \in \big[\frac{\alpha}{2}, 1-\frac{\alpha}{2}\big); \\ \mathcal{U}([-1,1)) & \text{if } Y \in \big[-\frac{\alpha}{2}, \frac{\alpha}{2}\big); \\ \mathcal{U}([-1,0)) & \text{if } Y \in \big[\frac{\alpha}{2}-1, -\frac{\alpha}{2}\big). \end{cases}$

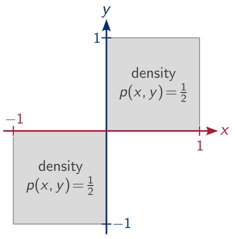
$$\Rightarrow h_P(X|Y) = \mathbb{E}_{y\sim P(Y)}\big[h_P(X\,|\,Y{=}y)\big] = \tfrac{1}{2}(1{-}\alpha)\log_2(1) + \alpha\log_2(2) + \tfrac{1}{2}(1{-}\alpha)\log_2(1) = \alpha.$$

- **Interpretation:** sign(X) has one bit of entropy, but a fraction α of possible observations of Y won't tell us sign(X) at all (while the other observations of Y tell it with certainty).

Symmary of Example 1

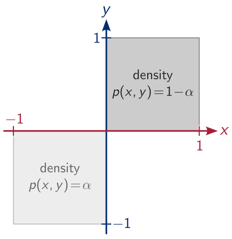
The mutual information $I_P(X;Y)$ takes into account:

Example 1a:



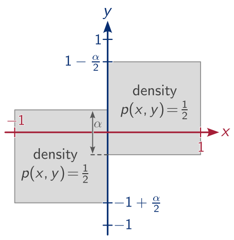
- how much new information an observation of X reveals about Y (and vice versa) …

Example 1b:



- … in comparison to how much we'd know about Y anyway;

Example 1c (Methods 1 & 2):



- with what *certainty* the new information is revealed; and
- how *probable* it is to make an informative observation.

Example 2: Gaussian Signal With Gaussian Noise

Consider an *analog* signal $x \sim \mathcal{N}(0, \sigma_s^2)$, sent over a *noisy* channel (e.g., voltage on a wire).

$\Longrightarrow$ Receiver receives a somewhat corrupted signal: $y \sim \mathcal{N}(x, \sigma_n^2)$.

Mutual information: $I_P(X;Y) = h_P(Y) - h_P(Y \,|\, X)$

- $p(y) = \mathbb{E}_{P(X)}\big[p(y \,|\, X)\big] = \int \mathcal{N}(x; 0, \sigma_s^2)\, \mathcal{N}(y; x, \sigma_s^2)\, \mathrm{d}x = \mathcal{N}(y; 0, \sigma_s^2 + \sigma_n^2)$

$\Longrightarrow$ $I_P(X;Y) = h_P(Y) - h_P(Y \,|\, X) = \frac{1}{2}\log_2(\sigma_s^2 + \sigma_n^2) - \frac{1}{2}\log_2(\sigma_n^2) = \frac{1}{2}\log_2\left(1 + \frac{\sigma_s^2}{\sigma_n^2}\right)$.

Interpretation: σ_s^2/σ_n^2 is the *signal-to-noise ratio* (SNR).

- For SNR $\to 0$, we have $I_P(X;Y) \to 0$; $\Longrightarrow$ receiver receives no meaningful information.
- But, as long as SNR > 0, one can still extract *some* information from the received signal.
- In the theory of channel coding (aka error correction), $P(Y \,|\, X)$ models a communication channel. Its *channel capacity* $C := \sup_{P(X)} I_P(X;Y)$ is the number of bits that can be transmitted noise-free per invocation of the noisy channel (in the limit of long messages).

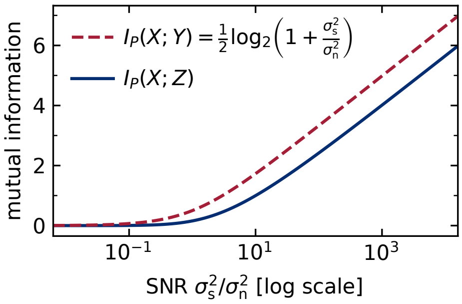
Data Processing Inequality I: Intuition

Remember when we were all still young and looking at slide 40?

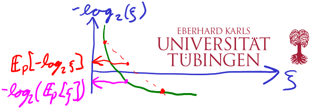
> $I_P(X;Y) = D_{\mathrm{KL}}\big(P(X,Y) \,\|\, P(X)\,P(Y)\big) = \mathbb{E}_P\left[-\log_2 \frac{p(X)\,p(Y)}{p(X,Y)}\right]$ (if densities p exist)
>
> - **Exercise:** let $X' = f(X)$, $Y' = g(Y)$, where f and g are differentiable *injective* functions. Convince yourself that I_P is independent of representation, i.e., $I_P(X';Y') = I_P(X;Y)$.

Question: what do *non-injective* transformations do to the mutual information?

- **Example:** start from last slide: $X \sim \mathcal{N}(0, \sigma_s^2)$; $Y|X \sim \mathcal{N}(X, \sigma_n^2)$.
- Then, consider $Z := Y^2$.
- Is $I_P(X;Z) \left\{\begin{array}{c}\textit{larger than,}\\ \textit{smaller than,}\\ \textit{or equal to}\end{array}\right\} I_P(X;Y)$?



Data Processing Inequality II: Formalization



Consider a **Markov chain**: $X \longrightarrow Y \longrightarrow Z$, i.e., $P(X,Y,Z) = P(X)\,P(Y|X)\,P(Z|Y)$.

$\Leftrightarrow$ X and Z are conditionally independent given Y (i.e., $P(X,Z \,|\, Y) = P(X|Y)\,P(Z|Y)$).

$\Leftrightarrow$ $Z \longrightarrow Y \longrightarrow X$ is a Markov chain (i.e., $P(X,Y,Z) = P(Z)\,P(Y|Z)\,P(X|Y)$).

Theorem (data processing inequality): *"once we've removed some information from a random variable, further processing cannot restore the removed information."*

- $\boxed{I_P(X;Y) \geq I_P(X;Z) \quad \text{and} \quad \underline{I_P(Y;Z) \geq I_P(X;Z)}}$ (∀ Markov chains $\underset{\shortparallel}{\cancel{X}\,P(Z|\cancel{Y})}\,Y \to Z$).

Proof: (discrete case)

$$I_P(Y;Z) - I_P(X;Z) = \mathbb{E}_P\left[-\log_2 \frac{P(Y)\,P(Z)}{P(Y,Z)} + \log_2 \frac{P(X)\,P(Z)}{P(X,Z)}\right] = \mathbb{E}_P\left[-\log_2 \frac{P(Y)\cancel{P(Z)}\,P(X,Z)}{P(Y,Z)\,P(X)\,\cancel{P(Z)}}\right]$$

$$\underset{\text{Jensen}}{\geq} -\log_2\left(\mathbb{E}_P\left[\frac{P(X,Z)}{P(Z|Y)\,P(X)}\right]\right) \underset{\text{Markov}}{=} -\log_2\left(\sum_{x,y,z} \cancel{P(X)}\,\underbrace{P(Y|X)}_{\to 1}\,\cancel{P(Z|Y)}\,\frac{P(X,Z)}{\cancel{P(Z|Y)}\,\cancel{P(X)}}\right) = -\log_2(1) = 0$$

Inf.-Theoretical Bounds on Model Performance

Consider a classification task: assign label Y to input data X: learn $P(Y\,|\,X)$

- Data generative distribution: $P(X, Y_{\text{g.t.}}) = P(Y_{\text{g.t.}})\,P(X\,|\,Y_{\text{g.t.}})$

  $\Longrightarrow$ Markov chain: $\boxed{Y_{\text{g.t.}} \xrightarrow{\text{data gen.}} X \xrightarrow{\text{classifier}} Y}$

  - Perfect classification would mean $Y = Y_{\text{g.t.}}$ $\Longrightarrow$ $I_P(Y_{\text{g.t.}}; Y) = \overbrace{H_P(Y_{\text{g.t.}})}^{\text{maximally possible mut. inf if the data source is fixed}} - \underbrace{H_P(Y_{\text{g.t.}}\,|\,Y)}_{=0}$
  - More generally: high accuracy $\Longrightarrow$ high $I_P(Y_{\text{g.t.}}; Y)$ $\Longrightarrow$ high $I_P(Y_{\text{g.t.}}; X) \geq I_P(Y_{\text{g.t.}}; Y)$:

    Bound: accuracy $\leq f^{-1}\big(I_P(Y_{\text{g.t.}}; X)\big)$ where $f(\alpha) = H_P(Y_{\text{g.t.}}) + \alpha \log_2 \alpha + (1-\alpha)\log_2 \frac{1-\alpha}{\#\text{classes}-1}$

    [Meyen, 2016 (MSc thesis advised by U. von Luxburg)]

- Now introduce a preprocessing step: $\boxed{Y_{\text{g.t.}} \xrightarrow{\text{data gen.}} X \xrightarrow{\text{preprocessing}} Z \xrightarrow{\text{classifier}} Y}$

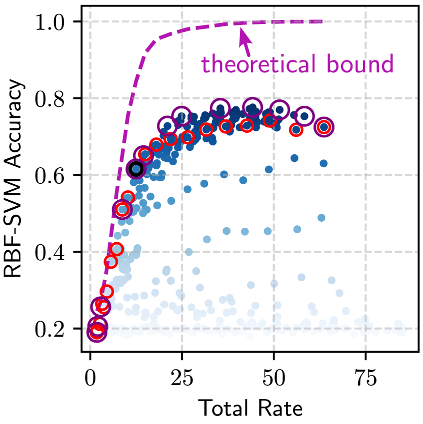
  - Theoretical bound now: accuracy $\leq f^{-1}\big(I_P(Y_{\text{g.t.}}; Z)\big) \leq f^{-1}\big(I_P(Y_{\text{g.t.}}; X)\big)$

    (by information processing inequality and monotonicity of f).

  $\Longrightarrow$ Information theory suggests: preprocessing can only hurt (bound on) downstream performance.

Limitations of Information Theory

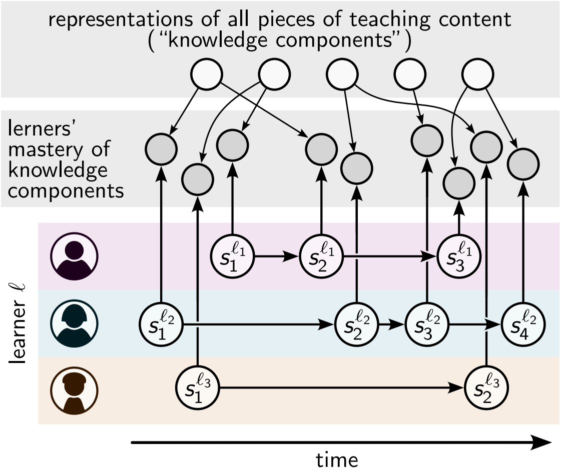
[Tim Xiao, RB, ICLR 2023]



- **Observation:** classification accuracy *decreases* for very large rate (= bound on mutual information).
- **Explanation:** information theory doesn't consider (computational/modeling) *complexity*.
  - Forcing the encoder to throw away some of the (least relevant) information can make downstream tasks *easier in practice*.
  - **Note:** it's the *information* bottleneck that can make downstream processing easier, not any (possible) dimensionality reduction.

    (In fact, many downstream tasks become *easier* in higher dimensions → kernel trick.)

Be Creative! You Now Have the Tools for It.



[Hanqi Zhou, RB, CM Wu, Á Tejero-Cantero, ICLR 2024]

We want to quantify:

- How **specific** are learner representations s for their learner ℓ?

  $I_P(s; \ell) = H_P(\ell) - H_P(\ell\,|\,s)$
- How **consistent** are representations for a fixed learner if we train on different subsets of time steps?

  $\mathbb{E}_{\ell_{\text{sub}}}\big[\,I_P(s^{\ell}; \ell_{\text{sub}})\,\big]$
- **Disentanglement**, i.e., how informative is each *component* of $s \in \mathbb{R}^n$ about learner identity ℓ?

  $H_P(s) - H_P(s\,|\,\ell)_{\text{diag}}$