Single-Shot Compression for Hypothesis Testing

Fabrizio Carpi, Siddharth Garg, Elza Erkip

New York University

22nd IEEE International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)

September 27-30, 2021
Outline

1. Introduction
2. Preliminaries
   - System Model
   - Hypothesis Testing
3. Hypothesis Testing Under Single-Shot Compression
4. Proposed Compressor
5. Results
   - Small alphabet $|\mathcal{X}| = 13$
   - Big alphabet $|\mathcal{X}| = 256$
6. Conclusion
Introduction

Preliminaries

Hypothesis Testing Under Single-Shot Compression

Proposed Compressor

Results

Conclusion
Motivation

A resource constrained **client** offloads costly task-related computations to a remote **server** (edge/cloud computing).

![Diagram of the data flow](image-url)
Motivation

A resource constrained **client** offloads costly task-related computations to a remote **server** (edge/cloud computing).

**Open question**: design **task-aware source coding** schemes which provide **effective** representations of the source data.
**In this paper**

**Assumptions**

- **Task**: binary hypothesis testing.
- **Client**: constrained device which cannot perform task locally, does not have memory and can only do simple scalar compression.
- **Server**: hypothesis testing on a block of compressed samples.
In this paper

Assumptions

- **Task**: binary hypothesis testing.
- **Client**: constrained device which cannot perform task locally, does not have memory and can only do simple scalar compression.
- **Server**: hypothesis testing on a block of compressed samples.

**Our work** → single-shot fixed-length compression for hypothesis testing.

- Problem formulation.
- Analyze the error performance.
- Propose a task-oriented compression algorithm for hypothesis testing.
Introduction

Preliminaries
- System Model
- Hypothesis Testing

Hypothesis Testing Under Single-Shot Compression

Proposed Compressor

Results

Conclusion
System Model

\( H_0 : X \sim P_0 \quad H_1 : X \sim P_1 \)

\( X \xrightarrow{\text{One-Shot Compressor}} \hat{X} \)  
\( \hat{X} \xrightarrow{\text{Buffer}} \hat{X}^n \)  
\( \hat{X}^n \xrightarrow{\text{Hypot. Test.}} \text{Accept or Reject } H_0 \)

\( X_1, \ldots, X_n \sim P_\theta \) are i.i.d. random variables.

---

<table>
<thead>
<tr>
<th>Source</th>
<th>Compressor</th>
<th>Hypothesis Testing</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x \in \mathcal{X} = {1, \ldots,</td>
<td>\mathcal{X}</td>
<td>} )</td>
</tr>
<tr>
<td>( X \sim P_\theta(x), \theta \in {0, 1} )</td>
<td>( \hat{X} = f(X), \hat{X} \sim \hat{P}_\theta(\hat{X}) )</td>
<td>( \hat{\theta} = 0 ) ( \hat{\theta} = 1 )</td>
</tr>
</tbody>
</table>

Fixed rate compression \( R = \log M \). We consider \( M < |\mathcal{X}| \).
Performance Metric

From classical binary hypothesis testing theory\(^1\):

1. if type-I error \(\epsilon < \epsilon\) \(\implies\) type-II error \(\beta_n^\epsilon\) decays exponentially in \(n\) as
   \[\gamma = -\lim_{n \to \infty} \frac{1}{n} \log \beta_n^\epsilon.\]
2. Chernoff-Stein Lemma (without compression): optimal type-II error exponent is
   \[\gamma^* = D(P_0 || P_1).\]

---


\(^2\)Type-II error: accept \(H_0\) when \(H_1\) is true.
Performance Metric

From classical binary hypothesis testing theory\(^1\):

1. if type-I error \( < \epsilon \implies \) type-II error \( \beta_n^\epsilon \) decays exponentially in \( n \) as 
   \[
   \gamma = - \lim_{n \to \infty} \frac{1}{n} \log \beta_n^\epsilon.
   \]
2. Chernoff-Stein Lemma (without compression): optimal type-II error exponent is 
   \[
   \gamma^* = D(P_0 || P_1).
   \]

Our performance metric \( \rightarrow \) type-II error exponent \( \gamma \).
With compression: the error exponent depends on \((f, R)\): \( \gamma_f(R) \).

---

\(^1\)Cover and Thomas, *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*.

\(^2\)Type-II error: accept \( H_0 \) when \( H_1 \) is true.
Performance Metric

From classical binary hypothesis testing theory\(^1\):

1. if type-I error \( < \epsilon \) \( \implies \) type-II error \( \beta_n^\epsilon \) decays exponentially in \( n \) as \( \gamma = - \lim_{n \to \infty} \frac{1}{n} \log \beta_n^\epsilon \).

2. Chernoff-Stein Lemma (without compression): optimal type-II error exponent is \( \gamma^* = D(P_0||P_1) \).

Our performance metric \( \rightarrow \) type-II error exponent \( \gamma \).

With compression: the error exponent depends on \((f, R)\): \( \gamma_f(R) \).

\( \implies \) We define the **compression penalty**: \( \Delta_f(R) = D(P_0||P_1) - \gamma_f(R) \).

---

\(^1\)Cover and Thomas, *Elements of Information Theory* (*Wiley Series in Telecommunications and Signal Processing*).

\(^2\)Type-II error: accept \( H_0 \) when \( H_1 \) is true.
1 Introduction

2 Preliminaries

3 Hypothesis Testing Under Single-Shot Compression

4 Proposed Compressor

5 Results

6 Conclusion
Hypothesis Testing on Compressed Variable

Lemma 1

The log-likelihood ratio test on the compressed variables $\hat{X}_i = f(X_i)$, $i = 1, \ldots, n$, is optimal; the corresponding optimal error exponent is $\gamma_f(R) = D(\hat{P}_0||\hat{P}_1)$.

Hence, the compression penalty is $\Delta_f(R) = D(P_0||P_1) - D(\hat{P}_0||\hat{P}_1)$. 
Hypothesis Testing on Compressed Variable

Lemma 1

The log-likelihood ratio test on the compressed variables \( \hat{X}_i = f(X_i), i = 1, \ldots, n, \) is optimal; the corresponding optimal error exponent is \( \gamma_f(R) = D(\hat{P}_0||\hat{P}_1). \)

Hence, the compression penalty is \( \Delta_f(R) = D(P_0||P_1) - D(\hat{P}_0||\hat{P}_1). \)

**Optimal compressor**: \( f^* = \arg \max_f D(\hat{P}_0||\hat{P}_1) = \arg \min_f \Delta_f \text{ s.t. } |f| \leq M. \)

NP-hard problem! Optimization over each possible \( f, \) which induces a partition of \( M \) sets over \( \mathcal{X}. \)
Hypothesis Testing Under Single-Shot Compression

Compression Penalty: $\Delta_f(R) = D(P_0 \parallel P_1) - D(\hat{P}_0 \parallel \hat{P}_1)$

Proposition 1

Expression for $\Delta_f \geq 0$:

$$\Delta_f = \sum_{\hat{x}=1}^{M} \hat{P}_0(\hat{x}) D(P_0(x|\hat{x}) \parallel P_1(x|\hat{x}))$$

where $P_\theta(x|\hat{x}) = \frac{P_\theta(x)}{\hat{P}_\theta(\hat{x})} \mathbb{1}\{\hat{x} = f(x)\}$ is the posterior of $X$ given $\hat{X} = f(X)$. 

Observations:

✓ The KL term is zero for one-to-one mappings (or if equal posteriors) → only the many-to-one mappings contribute to $\Delta_f(R)$.

→ In general, a good task-aware compression strategy combines $X$ values that have similar posteriors.
Compression Penalty: $\Delta_f(R) = D(P_0 \mid\mid P_1) - D(\hat{P}_0 \mid\mid \hat{P}_1)$

Proposition 1

Expression for $\Delta_f \geq 0$:

$$\Delta_f = \sum_{\hat{x}=1}^M \hat{P}_0(\hat{x}) D\left(P_0(x \mid \hat{x}) \mid\mid P_1(x \mid \hat{x})\right)$$

where $P_\theta(x \mid \hat{x}) = \frac{P_\theta(x)}{\hat{P}_\theta(\hat{x})} \mathbb{1}\{\hat{x} = f(x)\}$ is the posterior of $X$ given $\hat{X} = f(X)$.

Observations:

- The KL term is zero for one-to-one mappings (or if equal posteriors) $\rightarrow$ only the many-to-one mappings contribute to $\Delta_f(R)$.
- In general, a good task-aware compression strategy combines $X$ values that have similar posteriors.
1 Introduction

2 Preliminaries

3 Hypothesis Testing Under Single-Shot Compression

4 Proposed Compressor

5 Results

6 Conclusion
One-step compression from $|X|$ to $|X| - 1$

What is the optimal compressor when reducing the alphabet size by 1?
One-step compression from $|\mathcal{X}|$ to $|\mathcal{X}| - 1$

What is the optimal compressor when reducing the alphabet size by 1?

**Lemma 2**

One-step compression from $|\mathcal{X}|$ to $|\mathcal{X}| - 1$: $f$ combines $\{a, b\} \subset \mathcal{X}$ and the others $x \in \mathcal{X} \setminus \{a, b\}$ are one-to-one; i.e., $f(a) = f(b) = m \in \mathcal{M}$, $f(i) = i \in \mathcal{M} \setminus \{m\}$.

Then, the optimal compressor is

$$f^* = \arg \min_{\{a, b\} \subset \mathcal{X}: f(a) = f(b) = m} \left\{ \hat{P}_0(m) D \left( P_0(x|m) \parallel P_1(x|m) \right) \right\},$$

where $P_{\theta}(x|m) = \left[ \begin{array}{c} P_{\theta}(a) \\ P_{\theta}(a) + P_{\theta}(b) \end{array} \right], P_{\theta}(b) \left[ \begin{array}{c} P_{\theta}(b) \\ P_{\theta}(a) + P_{\theta}(b) \end{array} \right], \theta = \{0, 1\}$. 

F. Carpi, S. Garg, E. Erkip

Single-Shot Compression for Hypothesis Testing

SPAWC 2021 7 / 14
Our Proposed Compressor

Our “KL-greedy” compressor:

- iteratively reduce the alphabet size by 1 at each step, until the compressed alphabet has size $M$;
- at each step, combine $\{a, b\}$ which minimize Lemma 2;
- note that this compressor can be determined in polynomial time.
Proposed Compressor

**Our “KL-greedy” compressor:**

- iteratively reduce the alphabet size by 1 at each step, until the compressed alphabet has size $M$;
- at each step, combine $\{a, b\}$ which minimize Lemma 2;
- note that this compressor can be determined in polynomial time.

\[
P_0(x) = \begin{cases} 0.5 & \text{if } x = 1 \\ 0.25 & \text{if } x = 2 \\ 0.125 & \text{if } x = 3 \\ 0.125 & \text{if } x = 4 \end{cases}
\]

\[
P_1(x) = \begin{cases} 0.25 & \text{if } x = 1 \\ 0.125 & \text{if } x = 2 \\ 0.125 & \text{if } x = 3 \\ 0.5 & \text{if } x = 4 \end{cases}
\]

\[D(P_0 || P_1) = 0.5\]

Example: \[\hat{x} = f(x)\]
Our Proposed Compressor

Our “KL-greedy” compressor:

- iteratively reduce the alphabet size by 1 at each step, until the compressed alphabet has size $M$;
- at each step, combine $\{a, b\}$ which minimize Lemma 2;
- note that this compressor can be determined in polynomial time.

Example

\[
\begin{align*}
P_0(x) & \quad \quad P_1(x) \\
0.5 & \quad \quad 0.25 \\
0.25 & \quad \quad 0.125 \\
0.125 & \quad \quad 0.125 \\
0.125 & \quad \quad 0.5 \\
\hline
D(P_0||P_1) = 0.5
\end{align*}
\]

\[
\begin{align*}
x & \quad \quad \hat{x} = f(x) \\
1 \quad \quad \hat{P}_0(\hat{x}) \\
2 \quad \quad 0.75 \\
3 \quad \quad 0.125 \\
4 \quad \quad 0.125 \\
\hline
D(\hat{P}_0||\hat{P}_1) = 0.5
\end{align*}
\]
Our Proposed Compressor

Our “KL-greedy” compressor:

- iteratively reduce the alphabet size by 1 at each step, until the compressed alphabet has size $M$;
- at each step, combine $\{a, b\}$ which minimize Lemma 2;
- note that this compressor can be determined in polynomial time.

![Diagram showing the proposed compressor process and an example with probabilities and distances.

$D(P_0||P_1) = 0.5$ and $D(\hat{P}_0||\hat{P}_1) = 0.4564$]
Our Proposed Compressor

Our “KL-greedy” compressor:

- iteratively reduce the alphabet size by 1 at each step, until the compressed alphabet has size $M$;
- at each step, combine \{a, b\} which minimize Lemma 2;
- note that this compressor can be determined in polynomial time.

\[
P_0(x) \quad \quad P_1(x) \quad \quad x \quad \quad \hat{x} = f(x) \quad \hat{P}_0(x) \quad \hat{P}_1(x)
\]

\[
P_0(x) = 0.5 \\
\quad 0.25 \quad 0.25 \quad 0.125 \quad 0.125 \quad 0.125 \\
\quad 0.125 \\
\quad 0.5
\]

\[
P_1(x) = 0.25 \\
\quad 0.125 \quad 0.125 \quad 0.5 \\
\quad 0.5
\]

\[
D(P_0 || P_1) = 0.5
\]

\[
\Delta_f(1) = 0.0436
\]

\[
D(\hat{P}_0 || \hat{P}_1) = 0.4564
\]
Results

1 Introduction

2 Preliminaries

3 Hypothesis Testing Under Single-Shot Compression

4 Proposed Compressor

5 Results
   • Small alphabet $|\mathcal{X}| = 13$
   • Big alphabet $|\mathcal{X}| = 256$

6 Conclusion
Simulation details

$P_\theta$ are shifted binomial distributions with different parameters.

Compare compression penalty $\Delta_f$ and empirical type-II error rate for:

- optimal compressor $f^*$ — when feasible to compute, i.e, small $|\mathcal{X}|$;
- our KL-greedy compressor;
- universal compressor\(^3\) designed for reconstruction under log-loss distortion.

For the empirical type-II error rate, consider a threshold $T$ such that type-I error rate $< \epsilon = 0.05$ for a given compressor at rate $M$.

---

Distributions for $M = 4$

- $P_0(x)$ vs $P_1(x)$
- $\hat{P}_0(\hat{x})$ vs $\hat{P}_1(\hat{x})$

\[ \hat{x} = f(x) \quad \text{— Our compressor} \]

- $\hat{P}_0(\hat{x})$ vs $\hat{P}_1(\hat{x})$
- $\hat{x} = f(x) \quad \text{— Universal compressor} $

- The compressed KL is larger for our compressor (\textit{divergent} distributions, versus \textit{uniform} in the universal case).
- In our compressor: clustering of source symbols with same \textit{information}.
Compression Penalty $\Delta_f(R)$

- Our compressor performs close to the optimal.
- The compression penalty quickly approaches zero for increasing rate.
Type-II Error Rate for $n = 5, \epsilon = 0.05$

- Our compressor achieves error rate close to the uncompressed for increasing rate.
1 Introduction

2 Preliminaries

3 Hypothesis Testing Under Single-Shot Compression

4 Proposed Compressor

5 Results

6 Conclusion
Formulation for the optimal compressor for hypothesis testing (task-aware).
Proposed the empirical “KL-greedy” compressor → it can be computed in polynomial time and preserves the useful information.
Task-aware compression achieves error rate comparable to the uncompressed case for low rates.

Thank you! Q&A?

Single-Shot Compression for Hypothesis Testing

Fabrizio Carpi, Siddharth Garg, Elza Erkip
fabrizio.carpi@nyu.edu

22nd IEEE International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)
September 27-30, 2021

This work was supported in part by NSF–Intel grant #2003182 and NSF grant #1925079.