

# **Lossy compression for lossless prediction**

EECS Seminar: Advanced Topics in Machine Learning

---

Romain Graux

March 16, 2022

# Motivation

---

~ 50 trillion GB data collected per year

## Motivation

---

~ 50 trillion GB data collected per year

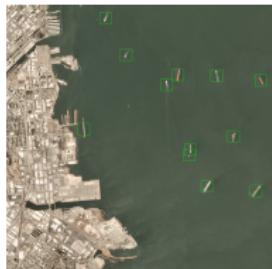
⇒ But most data is processed by algorithms performing **downstream tasks**.

# Motivation

---

~ 50 trillion GB data collected per year

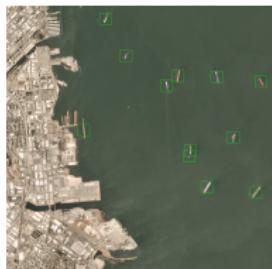
⇒ But most data is processed by algorithms performing **downstream tasks**.



# Motivation

~ 50 trillion GB data collected per year

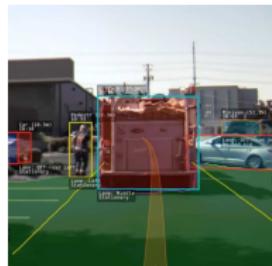
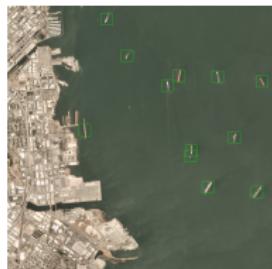
⇒ But most data is processed by algorithms performing **downstream tasks**.



# Motivation

~ 50 trillion GB data collected per year

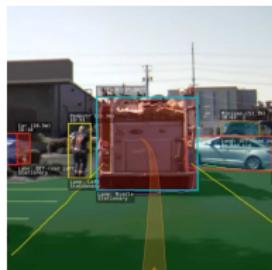
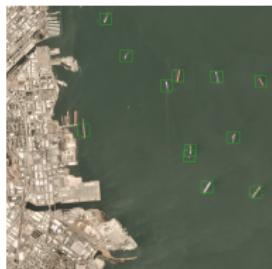
⇒ But most data is processed by algorithms performing **downstream tasks**.



# Motivation

~ 50 trillion GB data collected per year

⇒ But most data is processed by algorithms performing **downstream tasks**.



## Motivation

---

~ 50 trillion GB data collected per year

⇒ But most data is processed by algorithms performing **downstream tasks**.

Yet current compressors optimize high **perceptual** fidelity

# Motivation

---

~ 50 trillion GB data collected per year

⇒ But most data is processed by algorithms performing **downstream tasks**.

Yet current compressors optimize high **perceptual** fidelity

- Stores too much not needed information
- Does not ensure good task performance

# Motivation

---

~ 50 trillion GB data collected per year

⇒ But most data is processed by algorithms performing **downstream tasks**.

Yet current compressors optimize high **perceptual** fidelity

- Stores too much not needed information
- Does not ensure good task performance



Source

# Motivation

---

~ 50 trillion GB data collected per year

⇒ But most data is processed by algorithms performing **downstream tasks**.

Yet current compressors optimize high **perceptual** fidelity

- Stores too much not needed information
- Does not ensure good task performance



Source



High bitrate

# Motivation

~ 50 trillion GB data collected per year

⇒ But most data is processed by algorithms performing **downstream tasks**.

Yet current compressors optimize high **perceptual** fidelity

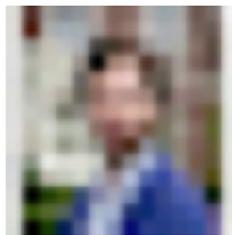
- Stores too much not needed information
- Does not ensure good task performance



Source



High bitrate



Low bitrate

# Motivation

~ 50 trillion GB data collected per year

⇒ But most data is processed by algorithms performing **downstream tasks**.

Yet current compressors optimize high **perceptual** fidelity

- Stores too much not needed information
- Does not ensure good task performance



Source



High bitrate



Low bitrate



Desired

## What they designed

---

They designed a **task-centric** distortion that ensures good downstream performance

## What they designed

---

They designed a **task-centric** distortion that ensures good downstream performance

- Characterize minimum bit-rate to ensure high performance on desired tasks;

## What they designed

---

They designed a **task-centric** distortion that ensures good downstream performance

- Characterize minimum bit-rate to ensure high performance on desired tasks;
- Derive unsupervised objectives for training **task-centric** compressors;

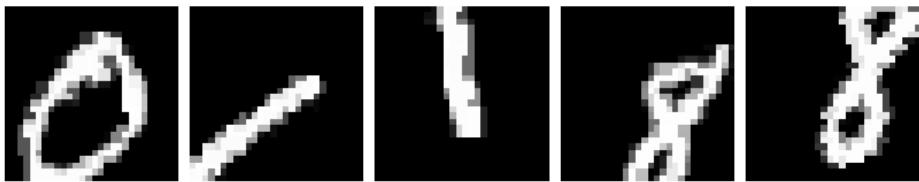
## What they designed

---

They designed a **task-centric** distortion that ensures good downstream performance

- Characterize minimum bit-rate to ensure high performance on desired tasks;
- Derive unsupervised objectives for training **task-centric** compressors;
- > 1000x compression gains on Imagenet compared to JPEG (see Slide 8).

# Intuition: Augmented MNIST

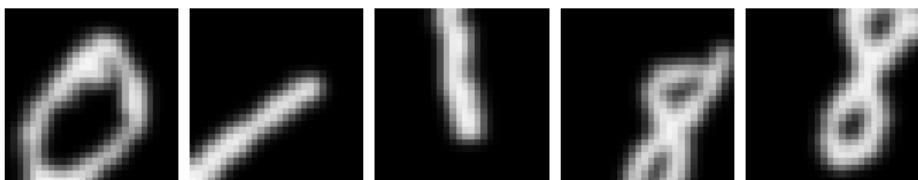


Source: Augmented MNIST

# Intuition: Augmented MNIST

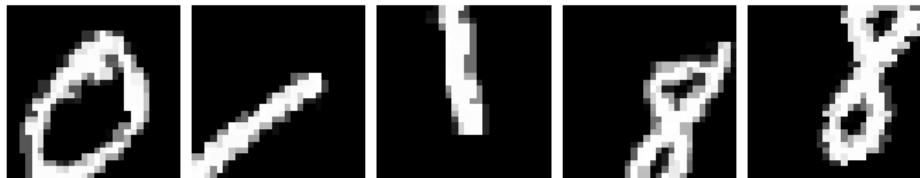


Source: Augmented MNIST

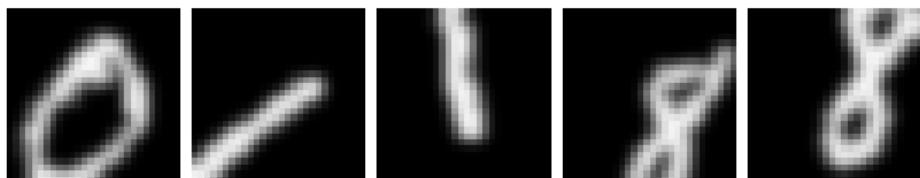


Standard neural compressor: 130 bit-rate

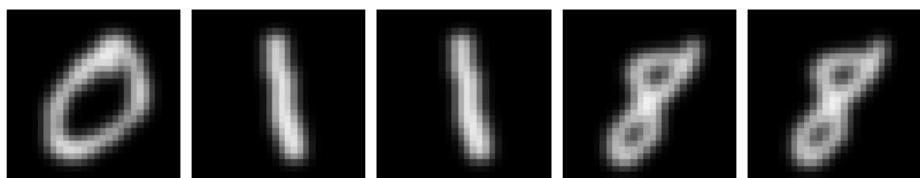
# Intuition: Augmented MNIST



Source: Augmented MNIST



Standard neural compressor: 130 bit-rate



Their neural compressor: 48 bit-rate

# Intuition: Augmented MNIST



**Prototypical** digit ensures

- high downstream performance
- good compression rate

# Intuition: Augmented MNIST



**Prototypical** digit ensures → high downstream performance  
→ good compression rate

Why not sending directly the labels?

# Intuition: Augmented MNIST



**Prototypical** digit ensures → high downstream performance  
→ good compression rate

Why not sending directly the labels?

- Might be interested in multiple downstream tasks;

# Intuition: Augmented MNIST



**Prototypical** digit ensures → high downstream performance  
→ good compression rate

Why not sending directly the labels?

- Might be interested in multiple downstream tasks;
- Would require knowing tasks of interest at compression time.

# Intuition: Augmented MNIST



**Prototypical** digit ensures → high downstream performance  
→ good compression rate

Why not sending directly the labels?

- Might be interested in multiple downstream tasks;
  - Would require knowing tasks of interest at compression time.
- ⇒ The objective is **unsupervised**

# Problem setup

---

**Goal:**

# Problem setup

---

## Goal:

- Minimum achievable bit-rate to store  $X$

# Problem setup

---

## Goal:

- Minimum achievable bit-rate to store  $X$ 
  - e.g.  $X$ : all satellite images

# Problem setup

---

## Goal:

- Minimum achievable bit-rate to store  $X$ 
  - e.g.  $X$ : all satellite images
  - e.g.  $X$ : all images of machine learning researchers

# Problem setup

---

## Goal:

- Minimum achievable bit-rate to store  $X$
- While ensuring high performance on any task  $Y$  of interest  
 $\tau = \{Y_1, Y_2, \dots\}$

# Problem setup

---

## Goal:

- Minimum achievable bit-rate to store  $X$
- While ensuring high performance on any task  $Y$  of interest  
 $\tau = \{Y_1, Y_2, \dots\}$ 
  - e.g.  $Y_1$ : how old is the person?

# Problem setup

---

## Goal:

- Minimum achievable bit-rate to store  $X$
- While ensuring high performance on any task  $Y$  of interest  
 $\tau = \{Y_1, Y_2, \dots\}$ 
  - e.g.  $Y_1$ : how old is the person?
  - e.g.  $Y_2$ : does the person wear glasses?

# Problem setup

---

## Goal:

- Minimum achievable bit-rate to store  $X$
- While ensuring high performance on any task  $Y$  of interest  
 $\tau = \{Y_1, Y_2, \dots\}$

⇒ looking for a representation  $Z$  s.t. predictions are approx. as good as using  $X$

# Problem setup

---

## Goal:

- Minimum achievable bit-rate to store  $X$
- While ensuring high performance on any task  $Y$  of interest  
 $\tau = \{Y_1, Y_2, \dots\}$

⇒ looking for a representation  $Z$  s.t. predictions are approx. as good as using  $X$

$$\sup_{Y \in \tau} \underbrace{R[Y|Z] - R[Y|X]}_{\text{excess Bayes risk}} \leq \delta$$

# Problem setup

---

## Goal:

- Minimum achievable bit-rate to store  $X$
- While ensuring high performance on any task  $Y$  of interest  
 $\tau = \{Y_1, Y_2, \dots\}$

⇒ looking for a representation  $Z$  s.t. predictions are approx. as good as using  $X$

$$\sup_{Y \in \tau} R[Y|Z] - R[Y|X] \underbrace{\leq}_{\text{small}} \delta$$

# Problem setup

---

## Goal:

- Minimum achievable bit-rate to store  $X$
- While ensuring high performance on any task  $Y$  of interest  
 $\tau = \{Y_1, Y_2, \dots\}$

⇒ looking for a representation  $Z$  s.t. predictions are approx. as good as using  $X$

$$\sup_{\substack{Y \in \tau \\ \text{all tasks}}} R[Y|Z] - R[Y|X] \leq \delta$$

# Problem setup

---

## Goal:

- Minimum achievable bit-rate to store  $X$
- While ensuring high performance on any task  $Y$  of interest  
 $\tau = \{Y_1, Y_2, \dots\}$

⇒ looking for a representation  $Z$  s.t. predictions are approx. as good as using  $X$

$$\sup_{Y \in \tau} R[Y|Z] - R[Y|X] \leq \delta$$

$\delta = 0$  : lossless prediction regime

# Problem setup

---

## Goal:

- Minimum achievable bit-rate to store  $X$
- While ensuring high performance on any task  $Y$  of interest  
 $\tau = \{Y_1, Y_2, \dots\}$

⇒ looking for a representation  $Z$  s.t. predictions are approx. as good as using  $X$

$$\sup_{Y \in \tau} R[Y|Z] - R[Y|X] \leq \delta$$

$\delta = 0$  : lossless prediction regime

**Problem:** Would assume access  $\tau$

## Key assumption: Invariance structure

---

Tasks of interest to humans share structure.

## Key assumption: Invariance structure

---

Tasks of interest to humans share structure.

**Assumption:** tasks of interest are invariant to some transformation

## Key assumption: Invariance structure

---

Tasks of interest to humans share structure.

**Assumption:** tasks of interest are invariant to some transformation

→ ubiquitous in ML: data augmentation

# Key assumption: Invariance structure

Tasks of interest to humans share structure.

**Assumption:** tasks of interest are invariant to some transformation

→ ubiquitous in ML: data augmentation

## Proposition

Exists a "worst task"  $M(X)$  that has all information to predict any variant task  $Y \in \tau$

$$R[M(X)|Z] = \sup_{Y \in \tau} R[Y|Z] - R[Y|X] \leq \delta$$

# Key assumption: Invariance structure

Tasks of interest to humans share structure.

**Assumption:** tasks of interest are invariant to some transformation

→ ubiquitous in ML: data augmentation

## Proposition

Exists a "worst task"  $M(X)$  that has all information to predict any variant task  $Y \in \tau$

$$R[M(X)|Z] = \sup_{Y \in \tau} R[Y|Z] - R[Y|X] \leq \delta$$

# Key assumption: Invariance structure

Tasks of interest to humans share structure.

**Assumption:** tasks of interest are invariant to some transformation

→ ubiquitous in ML: data augmentation

## Proposition

Exists a "worst task"  $M(X)$  that has all information to predict any variant task  $Y \in \tau$

$$R[M(X)|Z] = \sup_{Y \in \tau} R[Y|Z] - R[Y|X] \leq \delta$$

# Key assumption: Invariance structure

Tasks of interest to humans share structure.

**Assumption:** tasks of interest are invariant to some transformation

→ ubiquitous in ML: data augmentation

## Proposition

Exists a "worst task"  $M(X)$  that has all information to predict any variant task  $Y \in \tau$

$$R[M(X)|Z] = \sup_{Y \in \tau} R[Y|Z] - R[Y|X] \leq \delta$$

# Worst task example

---



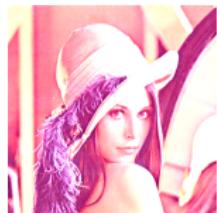
$x_0$ : gray scale

# Worst task example

---



$x_0$ : gray scale



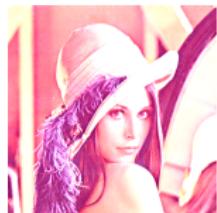
$x_1$ : brightness

# Worst task example

---



$x_0$ : gray scale



$x_1$ : brightness

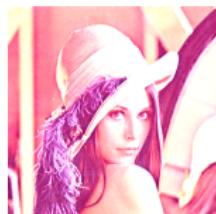


$x_2$ : horizontal flip

# Worst task example



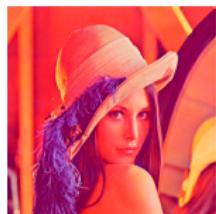
$x_0$ : gray scale



$x_1$ : brightness

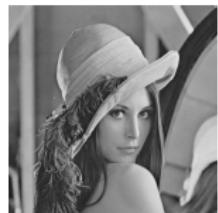


$x_2$ : horizontal flip

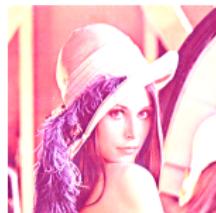


$x_3$ : saturation

# Worst task example



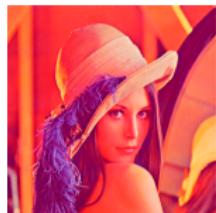
$x_0$ : gray scale



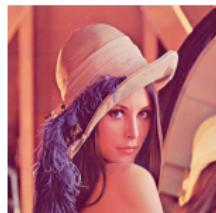
$x_1$ : brightness



$x_2$ : horizontal flip



$x_3$ : saturation



$M(x)$ : unaugmented

# Performance