ON THE LIMITS OF MAXIMAL CODING RATE REDUCTION FOR OUT-OF-DISTRIBUTION GENERALISATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Substantial efforts have been devoted to making deep learning objectives, representations, and architectures interpretable, with the goal of improving the safety, robustness, and generalisation of learning systems in diverse real-world applications. The recently proposed maximal coding rate reduction (MCR^2) offers a promising information-theoretic framework for learning structured, discriminative representations of class-wise submanifolds and has inspired interpretable white-box architectures. However, we observe that MCR^2 can completely fail under distribution shift, motivating our study of its out-of-distribution (OOD) generalisation limits. We establish two limitations of MCR^2 for OOD generalisation. First, the MCR^2 objective alone can admit complete prediction failure: a representation based entirely on unstable environmental features can achieve the global coding optimum yet fail completely after correlation reversal, despite an available perfectly stable feature. This exact-optimum example includes test inputs that cannot occur during training. Even when every possible test input can also occur during training, coding quality can be arbitrarily close to optimal while prediction error is arbitrarily close to 100%. Second, directly incorporating the invariance principle underlying widely successful invariant risk minimisation (IRM) and risk extrapolation (REx) does not eliminate this failure. The failing representation admits the same optimal coding operator across training environments, showing that shared coding optimality does not ensure stable prediction. Reliable OOD guarantees for MCR^2 therefore require additional new assumptions or learning principles that establish stable predictive relationships across environments.

1 INTRODUCTION

Over the past decade, deep learning has transformed a wide range of applications, from language processing (Devlin et al., 2019) and computer vision (He et al., 2016) to scientific discovery (Abramson et al., 2024) and robotics (Kaufmann et al., 2023). Despite these advances, modern deep networks are still designed largely through trial and error (Elsken et al., 2019; Zhou et al., 2024). Researchers often choose architectures and training strategies through extensive experimentation (Zoph & Le, 2017; Hutter et al., 2019). Yet it remains difficult to explain the resulting systems through a clear set of computational principles (Lipton, 2018). This gap motivates the development of interpretable learning systems (Rudin, 2019) whose objectives, representations, and computations can be understood within a common mathematical framework (Ma et al., 2022).

Maximal coding rate reduction (MCR^2) offers a promising approach to this goal (Yu et al., 2020). It starts with a basic question about representation learning: *what structure should a network learn and preserve from data?* The framework assumes that high-dimensional data from each class often lie near a low-dimensional submanifold (Ma et al., 2022). A useful representation should capture this class-specific structure, preserve meaningful variation within each class, and separate different classes. Building on earlier work that connects lossy data coding with data segmentation (Ma et al., 2007), MCR^2 introduces an information-theoretic objective that maximises the difference between the coding rate of the complete dataset and the average coding rate of its individual classes. This objective gives a precise way to describe representation geometry: it encourages compact class representations, diversity within each class, and separation between classes (Yu et al., 2020; Chan et al., 2022; Wang et al., 2024; Yu et al., 2024).

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

The importance of MCR^2 extends from representation geometry to principled network design. Its geometric analyses describe solutions in which classes occupy mutually orthogonal subspaces and retain variation within each subspace (Yu et al., 2020; Wang et al., 2024). This connects a computable objective value to properties of the learned representation. Crucially, coding-rate principles also provide a way to derive white-box architectures whose layers have explicit mathematical roles. ReduNet unfolds the optimisation of MCR^2 into network layers with optimisation, statistical, and geometric interpretations (Chan et al., 2022). Building on the coding-rate perspective, CRATE derives Transformer-like architectures from a sparse rate-reduction objective, giving attention and subsequent layers explicit roles in compressing and sparsifying representations (Yu et al., 2024). These developments connect learning objectives, representation structure, and network computation within a common framework, making coding-rate principles relevant to the broader goal of interpretable deep learning. Coding-rate principles have also been extended to incremental learning (Wu et al., 2021; Tong et al., 2023), joint discriminative and generative learning (Dai et al., 2022; Tong et al., 2024), image clustering (Chu et al., 2024), and self-supervised representation learning (Wu et al., 2025). Their influence on both representation learning and architecture design makes it important to understand what the coding objective itself can guarantee about prediction reliability.

Despite these advantages, we find in practice that representations learned with MCR^2 can lead to severe out-of-distribution (OOD) prediction failures. On the Waterbirds dataset (Sagawa et al., 2020a), an encoder trained with MCR^2 performs poorly on test groups whose background–label associations oppose those prevalent during training (Table 1). The saliency maps in Figure 3 further illustrate its sensitivity to background regions. These failures raise concerns about the reliability of MCR^2 representations in real-world deployment and provide the main motivation for examining what the objective can guarantee under distribution shift.

We first analyse the OOD generalisation of MCR^2 and show that optimal coding geometry alone does not guarantee reliable prediction in a new environment. We use a simple two-class model to isolate the failure mechanism and demonstrate that this limitation arises even in a basic classification setting. An encoder using only an environmental feature can achieve the global coding optimum, despite the availability of a perfectly reliable feature. When the environmental feature's relationship with the label reverses, the classes remain well separated, but the source-optimal classifier predicts every test label incorrectly. This noiseless example includes test inputs that cannot occur during training. We further show that even when every possible test input can also occur during training, coding quality can approach the global optimum while prediction error approaches 100%.

Given the success of the invariance principle in OOD learning approaches such as invariant risk minimisation (IRM; Arjovsky et al., 2019) and risk extrapolation (REx; Krueger et al., 2021), a natural next step is to examine whether this principle can overcome the limitation of MCR^2. We show that even directly incorporating this principle into the coding-rate framework does not eliminate OOD prediction failure. In this formulation, invariance requires the same coding operator to be optimal in every training environment. Yet a representation relying entirely on an unstable environmental feature satisfies this requirement and still fails when the feature's relationship with labels reverses.

To the best of our knowledge, this work provides the first systematic analysis of the limitations of influential MCR^2 framework for OOD generalisation. Our main contributions are as follows:

- **The MCR^2 objective alone does not guarantee reliable OOD generalisation.** We first prove that MCR^2 can fail completely in OOD generalisation even in a simple noiseless two-class model. An encoder relying solely on an environmental feature achieves the global MCR^2 optimum, despite the availability of a perfectly stable predictive feature. When the envirnmental correlation reverses, its source-optimal classifier misclassifies every test example. This construction includes test inputs that cannot occur during training.

  We then extend the analysis to noisy environments where training and test data share the same possible inputs. By deriving the exact coding-rate reduction, prediction risk, and global coding optimum, we show that an arbitrarily small gap from the MCR^2 optimum can coexist with almost complete prediction failure, for any fixed coding precision and any finite number of training environments.

- **Directly incorporating the invariance principle into MCR^2 does not eliminate OOD failure.** Motivated by the success of invariance-based approaches such as IRM and REx, we

directly incorporate this principle into the MCR^2 framework through invariant coding rate reduction (ICR^2), which requires the same coding operator to be optimal across training environments. We prove that an encoder relying solely on an unstable environmental feature satisfies this requirement exactly, yet its source-optimal classifier can still misclassify almost every test example after correlation reversal. This failure persists even when training and test data share the same possible inputs and the coding value is arbitrarily close to the constrained global optimum.

These results distinguish the ability to organise class representations from the ability to predict reliably across environments. For representations learned with MCR^2, these results identify the need for new assumptions or learning principles that address stable predictive relationships beyond the coding objective and the shared-optimality requirement studied here.

2 METHOD

We first review the MCR^2 objective and use an object-and-background example and existing Waterbirds results to explain the motivation for our analysis. We then define the population objective, prediction risk, and two-class model. Vectors and matrices are written in bold throughout the paper.

2.1 THE MAXIMAL CODING RATE REDUCTION OBJECTIVE

Consider a dataset $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n] \in \mathbb{R}^{D\times n}$ with n samples from K classes. Representation learning maps these high-dimensional observations to lower-dimensional features that retain useful information for tasks such as classification and generation. The *manifold hypothesis* suggests that samples from each class concentrate near a low-dimensional submanifold $\mathcal{M}_y \subset \mathbb{R}^D$, so the full dataset lies near their union $\mathcal{M} = \bigcup_{y=1}^{K} \mathcal{M}_y$ (Hein & Audibert, 2005; Pope et al., 2021; Wright & Ma, 2022; Ma et al., 2022). Building on this hypothesis, the MCR^2 learning objective aims to capture the structure of each class submanifold and organise the resulting representations into mutually orthogonal low-dimensional subspaces (Yu et al., 2020). To learn such representations, a nonlinear encoder $f_{\boldsymbol{\Theta}} : \mathbb{R}^D \to \mathbb{R}^d$ maps each input $\mathbf{x}_i$ to a feature vector $\mathbf{z}_i = f_{\boldsymbol{\Theta}}(\mathbf{x}_i)$. Collecting these features gives the representation matrix $\mathbf{Z} = [\mathbf{z}_1, \dots, \mathbf{z}_n] \in \mathbb{R}^{d\times n}$, typically with $d \ll D$.

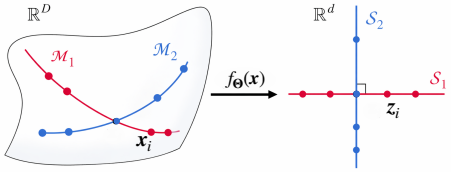

Figure 1: Class-wise submanifolds mapped to orthogonal subspaces by MCR^2. Adapted from Figure 1 of Yu et al. (2020).

In the supervised setting, let $\boldsymbol{\Pi}_y \in \mathbb{R}^{n\times n}$ be a diagonal membership matrix whose ith diagonal entry is one if sample i belongs to class y, and zero otherwise. The number of samples in class y is $n_y = \operatorname{tr}(\boldsymbol{\Pi}_y) > 0$. The learning objective of MCR^2 is maximising the coding-rate reduction:

$$\widehat{J}(\mathbf{Z}) = \underbrace{\frac{1}{2}\log\det\big(\mathbf{I}_d + \tfrac{d}{n\epsilon^2}\mathbf{Z}\mathbf{Z}^\top\big)}_{\text{Expansion: } \mathcal{R}(\mathbf{Z})} - \underbrace{\sum_{y=1}^{K}\tfrac{n_y}{2n}\log\det\Big(\mathbf{I}_d + \tfrac{d}{n_y\epsilon^2}\mathbf{Z}\boldsymbol{\Pi}_y\mathbf{Z}^\top\Big)}_{\text{Compression: } \mathcal{R}_c(\mathbf{Z},\boldsymbol{\Pi})}, \tag{1}$$
$$\text{subject to} \quad \|\mathbf{z}_i\|_2 = 1, \qquad i = 1, \dots, n.$$

where $\epsilon > 0$ is the coding precision. The normalisation constraint ensures a common feature scale, allowing fair comparisons of coding-rate reduction across representations.

The expansion term $\mathcal{R}(\mathbf{Z})$ encourages the features to spread out in the feature space. The compression term $\mathcal{R}_c(\mathbf{Z}, \boldsymbol{\Pi})$ encourages the features of each class to occupy a compact subspace. Together, this learning objective encourages the features of each class to lie in a low-dimensional linear subspace, with mutually orthogonal subspaces across classes (Yu et al., 2020), as illustrated in Figure 1. At an orthogonal solution,

$$(\mathbf{Z}\boldsymbol{\Pi}_y)^\top(\mathbf{Z}\boldsymbol{\Pi}_{y'}) = \mathbf{0}, \qquad y \neq y'. \tag{2}$$

2.2 Concerns about learning environmental features

Although MCR^2 encourages structured class representations, the objective does not specify which input factors the encoder should use. Both object features and background features can have low-dimensional structure and can help separate the training classes.

Our intuition comes from the following example. Consider the camel image in Figure 2. Both the object and the background lie on low-dimensional submanifolds, denoted by $\mathcal{M}_{\mathrm{obj}}$ and $\mathcal{M}_{\mathrm{back}}$, respectively. If capturing $\mathcal{M}_{\mathrm{back}}$ yields a higher coding-rate reduction than capturing $\mathcal{M}_{\mathrm{obj}}$, the MCR^2 objective favours the background representation. The model can therefore achieve strong class separation during training by learning background cues. When the background–label relationship changes, however, a classifier relying on these cues may fail.

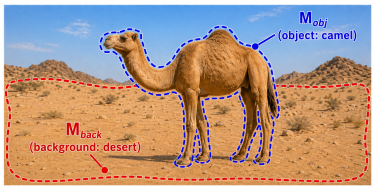

Figure 2: Object and background structures in a camel image.

Evidence from Waterbirds. To examine whether this concern arises in practice, we train a ResNet-18 encoder (He et al., 2016) from scratch using MCR^2 on Waterbirds (Sagawa et al., 2020a). The task is to distinguish landbirds ($y = 0$) from waterbirds ($y = 1$). Each image also has a land ($p = 0$) or water ($p = 1$) background. In the training set, 95% of waterbirds appear on water backgrounds and 95% of landbirds appear on land backgrounds; the remaining 5% of each class have the opposite background. Thus, the background is strongly associated with the bird label during training. The experimental setup is described in Appendix E.

Table 1: Accuracy (%) on the two background-mismatched Waterbirds test groups.

Overall	$(y = 0, p = 1)$	$(y = 1, p = 0)$
46.84	57.07	10.90

We evaluate the encoder on landbirds with water backgrounds ($y = 0, p = 1$; 2,255 images) and waterbirds with land backgrounds ($y = 1, p = 0$; 642 images), reversing the predominant training background–label association. Table 1 reports per-group and combined accuracy. Combined accuracy is 46.84%, below the 50% expected from uniform random guessing, while accuracy on waterbirds with land backgrounds is only 10.90%, indicating severe failure under this shift. Saliency maps for one training image per class show stronger sensitivity to background regions than to much of the bird (Figure 3). Together with the test results, they suggest reliance on background cues, illustrating that structured representations learned by MCR^2 alone do not ensure stable prediction across environments.

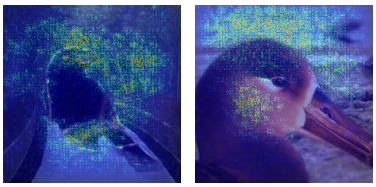

Figure 3: MCR^2 saliency maps on Waterbirds.

These experimental observations motivate a theoretical investigation of the limitations of MCR^2 under distribution shift. To study these limitations beyond the specific Waterbirds setting, we analyse a model with a perfectly stable feature and an environmental feature, allowing us to calculate the coding objective and prediction risk exactly and examine whether coding quality guarantees OOD prediction reliability.

2.3 Population formulation and prediction risk

In this section, we first express the MCR^2 objective in terms of data distributions and define prediction risk, allowing us to compare the objective value with classification performance across environments. We then introduce a model containing both a stable feature and an environmental feature whose association with the label can change. This model lets us study whether the objective distinguishes representations that support stable prediction from those that rely on environmental associations.

Population MCR^2 objective. Let P_e denote the distribution of $(\mathbf{X}, Y)$ in environment e, with class probabilities $p_y^e = P_e(Y = y)$. For an encoder $f : \mathcal{X} \to \mathbb{S}^{d-1}$, define the uncentred second moments

$$\mathbf{M}_{e,y}(f) = \mathbb{E}_e[f(\mathbf{X})f(\mathbf{X})^\top \mid Y = y], \qquad \mathbf{M}_e(f) = \sum_y p_y^e \mathbf{M}_{e,y}(f). \tag{3}$$

Replacing the empirical moments in Equation (1) with these expectations gives the population MCR^2 objective:

$$J_e(f) = \frac{1}{2}\log\det(\mathbf{I}_d + a\mathbf{M}_e(f)) - \frac{1}{2}\sum_y p_y^e \log\det(\mathbf{I}_d + a\mathbf{M}_{e,y}(f)), \qquad a = \frac{d}{\epsilon^2}. \tag{4}$$

Concavity of $\log\det$ ensures $J_e(f) \geq 0$.

Learning and evaluation. For a finite set $\mathcal{E}_{\mathrm{tr}}$ of training environments with weights $w_e > 0$ satisfying $\sum_{e\in\mathcal{E}_{\mathrm{tr}}} w_e = 1$, define

$$J_{\mathrm{tr}}(f) = \sum_{e\in\mathcal{E}_{\mathrm{tr}}} w_e J_e(f), \qquad \mathcal{L}_e(g\circ f) = P_e(g(f(\mathbf{X})) \neq Y). \tag{5}$$

Here, J_{tr} is the training MCR^2 objective and $\mathcal{L}_e$ is the classification error in environment e. For each representation, we use a source-optimal classifier that minimises $\sum_{e\in\mathcal{E}_{\mathrm{tr}}} w_e\mathcal{L}_e(g\circ f)$ over all classifiers and remains fixed during testing. This isolates representation quality from classifier estimation or optimisation errors. For both encoders below, a linear decision rule attains this optimum.

A stable factor and an environmental factor. Our main results consider $d = 2$, binary labels, and inputs $\mathbf{X} = (C, S) \in \{0,1\}^2$, where

$$Y \sim \mathrm{Bernoulli}(1/2), \qquad C = Y, \qquad S = Y \oplus B_e, \qquad B_e \sim \mathrm{Bernoulli}(\delta_e), \quad B_e \perp Y. \tag{6}$$

Here, $\oplus$ denotes exclusive-or. The stable feature C predicts the label perfectly in every environment, whereas the environmental feature S disagrees with it with probability δ_e. These are predictive relationships and require no causal assumptions.

We consider all unit-norm encoders

$$\mathcal{F} = \{f : \{0,1\}^2 \to \mathbb{S}^1\}. \tag{7}$$

With $\mathbf{b}_0 = (1,0)^\top$ and $\mathbf{b}_1 = (0,1)^\top$, define

$$f_C(c,s) = \mathbf{b}_c, \qquad f_S(c,s) = \mathbf{b}_s, \qquad g_0(\mathbf{z}) = \mathbf{1}\{z_2 > z_1\}. \tag{8}$$

The encoders f_C and f_S use the stable and environmental features, respectively, mapping each to the same two orthogonal unit vectors. Their input probabilities are

$$P_e(C = c, S = s) = \tfrac{1}{2}\big[(1-\delta_e)\mathbf{1}\{c = s\} + \delta_e\mathbf{1}\{c \neq s\}\big]. \tag{9}$$

For $0 < \delta_e < 1$, all four inputs have positive probability. Varying δ_e therefore changes the environmental association while preserving input support, allowing us to compare the two encoders' MCR^2 objective values and prediction risks under this shift.

3 Discriminative coding geometry does not ensure OOD generalisation

We show how a representation can have a coding value close to the best possible value while a classifier trained on it predicts almost every target label incorrectly. We first identify the global coding optimum and then compare it with the environmental encoder. In this section $a = 2/\epsilon^2 > 0$ is fixed.

Lemma 3.1 (Sharp binary coding bound)**.** *For any balanced binary distribution and any unit-norm encoder into $\mathbb{R}^2$,*

$$J_e(f) \leq J_\star(a) := \log(1 + a/2) - \tfrac{1}{2}\log(1 + a). \tag{10}$$

Equality holds if and only if the two class-conditional second moments are rank-one projectors onto orthogonal directions. In model (6), f_C attains equality in every environment, so

$$\max_{f\in\mathcal{F}} J_{\mathrm{tr}}(f) = J_\star(a).$$

Lemma 3.1 gives a precise meaning to successful coding geometry in this model: each class occupies one direction, and the two directions are orthogonal. The stable encoder achieves this structure in every environment, so optimal coding and reliable prediction are both possible. The lemma also provides a global benchmark over the entire encoder class. The next proposition measures how closely a representation based only on the environmental feature can approach it.

Proposition 3.2 (Exact environmental objective and prediction risk)**.** *Under Equation (6), define*

$$\Gamma_a(\delta) := \frac{1}{2}\log\left(1 + \frac{a^2}{1+a}\delta(1-\delta)\right). \tag{11}$$

Then $J_e(f_S) = J_\star(a) - \Gamma_a(\delta_e)$ *and* $\mathcal{L}_e(g_0 \circ f_S) = \delta_e$*. If* $\bar{\delta} := \sum_e w_e\delta_e < 1/2$*,* g_0 *is the unique source-optimal decision rule on the two output codes. In contrast,* $\mathcal{L}_e(g_0 \circ f_C) = 0$ *for every* e*.*

When δ_e is small, the environmental feature usually agrees with the label. It therefore produces both a small coding gap and a small source prediction error. Yet the classifier has learned to read that feature, whose reliability can change across environments. Proposition 3.2 makes the consequence explicit: the classifier's error in any environment is exactly the probability that the environmental feature disagrees with the label. The next result shows how large the gap between coding quality and target reliability can become.

Theorem 3.3 (Arbitrarily small coding gap and almost maximal target error)**.** *Fix* $a > 0$*, a positive integer* E*, positive source weights summing to one, and a tolerance* $\eta > 0$*. There exist* E *training environments and a target environment from Equation (6), all with the same input support, such that*

$$0 \le \max_{f\in\mathcal{F}} J_{\mathrm{tr}}(f) - J_{\mathrm{tr}}(f_S) \le \eta, \qquad \mathcal{L}_{\mathrm{te}}(g_0 \circ f_S) \ge 1 - \eta, \tag{12}$$

while g_0 *is source-optimal on* f_S *and* $\mathcal{L}_{\mathrm{te}}(g_0 \circ f_C) = 0$*. The source mismatch probabilities can be chosen distinct. More explicitly, if* $0 < \delta_e \le \delta_{\max} < 1/2$*, then*

$$J_\star(a) - J_{\mathrm{tr}}(f_S) = \sum_e w_e\Gamma_a(\delta_e) \le \frac{a^2}{2(1+a)}\delta_{\max}. \tag{13}$$

Choosing $\delta_{\mathrm{te}} = 1 - \delta_{\max}$ *yields target error* $1 - \delta_{\max}$*.*

Meaning and scope. Theorem 3.3 shows that an arbitrarily small coding gap can coexist with an almost maximal target error. Every possible target input already has positive source probability. What changes is the frequency of inputs on which the environmental feature predicts the wrong label: these inputs are rare during training and common at test time. Observing the same possible inputs therefore does not by itself ensure reliable prediction when their probabilities change.

The distinction between exact and approximate optimisation matters here. At positive noise, f_C has a strictly higher objective than f_S. More strongly, in this finite model with full source support, every exact global optimiser yields zero target error with an unrestricted source-optimal classifier (Appendix B.4). The theorem establishes that an arbitrarily small objective gap does not provide a uniform OOD guarantee over the permitted shifts. It does not assert failure of exact optimisers in this setting or exclude guarantees that additionally control the size of the shift.

At the noiseless boundary, one obtains an exact, but support-changing, statement. If every source has $\delta_e = 0$, both f_C and f_S globally maximise the source objective. At a target with $\delta_{\mathrm{te}} = 1$, the source-optimal classifier on f_S has error one. This example establishes complete prediction failure at an exact optimum with changing input support, whereas Theorem 3.3 establishes near-optimality and arbitrarily severe failure under identical input support.

3.1 Coding quality can survive correlation reversal

Note that the failure is not necessarily accompanied by a deterioration of coding geometry on the target distribution. From Equation (11),

$$J_\delta(f_S) = J_{1-\delta}(f_S), \qquad \mathcal{L}_\delta(g_0 \circ f_S) = \delta, \qquad \mathcal{L}_{1-\delta}(g_0 \circ f_S) = 1 - \delta. \tag{14}$$

For example, if the environmental cue agrees with the label in 99% of source samples, the source classifier has error 1%. After reversal, its error becomes 99%, while the coding value remains exactly the same. The representation still distinguishes the classes, but the source classifier uses the wrong label assignment for most target inputs.

Both environments have the same marginal representation law, namely the uniform distribution on $\{\mathbf{b}_0, \mathbf{b}_1\}$. Reversal swaps the two class-conditional moment matrices, leaving their average and their weighted log-determinants unchanged. The target remains easy to classify *if a new target head is fitted*; its Bayes head on f_S assigns the opposite labels. The task here is to use the source head without such refitting.

Figure 4 illustrates the noiseless limit. In the positive-noise case, each class occupies both codes, with a large mass on one and a small mass on the other. Thus exact class-subspace orthogonality holds at the limit, whereas the identical-support theorem establishes near-optimality with small cross-class contamination. We make no claim of exact orthogonality at positive noise. Also, target coding-rate reduction uses target labels in Equation (4); it is a theoretical diagnostic, not an unlabelled deployment statistic.

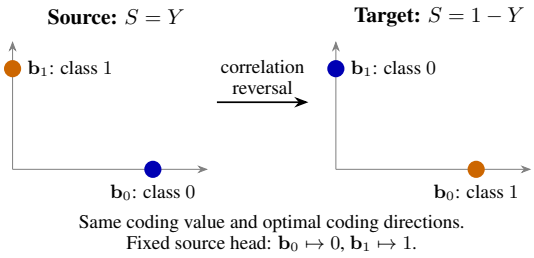

Figure 4: Optimal coding geometry survives correlation reversal; prediction fails.

4 Shared Optimal Coding Does Not Ensure Generalisation

The preceding results show why good coding geometry alone does not ensure stable prediction. Given the success of invariance-based methods such as invariant risk minimisation (IRM) (Arjovsky et al., 2019) and risk extrapolation (REx) (Krueger et al., 2021) in OOD generalisation, it is natural to ask whether incorporating the invariance principle into MCR^2 can address the failures established above. We show, however, that a direct formulation based on shared optimal coding can still admit severe OOD prediction failure.

Specifically, IRM and REx impose invariance at the level of prediction: IRM seeks a classifier that is simultaneously optimal across training environments, while REx penalises differences in prediction risk across environments. Within the MCR^2 framework, the learning criterion evaluates representation geometry through coding-rate reduction. A direct analogue of shared classifier optimality therefore requires a common linear coding operator that transforms the representation to maximise the MCR^2 objective in every training environment. For convenience, we refer to this shared-optimality requirement as invariant coding rate reduction (ICR^2). This constraint enforces agreement on optimal coding structure across environments but, as our analysis shows, does not ensure stable predictive relationships.

4.1 A Shared-Optimal-Coding Constraint

The coding operator acts on the learned representation and changes the directions assigned to its outputs. If different environments favoured incompatible operators, a shared-optimality requirement could reject the representation. The issue is whether agreement on an operator also ensures that a classifier can keep using the same label assignments. For $f \in \mathcal{F}$, define the set of admissible linear coding operators by

$$\mathcal{D}(f) = \{\mathbf{D} \in \mathbb{R}^{2\times 2} : \|\mathbf{D}f(\mathbf{x})\|_2 = 1 \text{ for all } \mathbf{x} \in \mathcal{X}\}. \tag{15}$$

Here $\mathcal{X} = \{0,1\}^2$ is fixed, and $\mathbf{I}_2 \in \mathcal{D}(f)$. The constraint applies to the encoded representations $\mathbf{D}f(\mathbf{x})$ rather than to the operator's Frobenius norm. For f_C and f_S, which attain both basis codes, this set is exactly

$$\mathcal{D}(f_C) = \mathcal{D}(f_S) = \{[\mathbf{u}, \mathbf{v}] : \mathbf{u}, \mathbf{v} \in \mathbb{S}^1\}. \tag{16}$$

It includes non-orthogonal columns and rank-one operators; it is not restricted to rotations for which rate reduction is automatically unchanged.

Consider the following idealised invariant coding-rate reduction objective:

$$\begin{aligned} J^*_{\mathrm{ICR}} = \sup_{f\in\mathcal{F},\, \mathbf{D}\in\mathcal{D}(f)} \quad & \sum_{e\in\mathcal{E}_{\mathrm{tr}}} w_e J_e(\mathbf{D}f) \\ \text{subject to} \quad & \mathbf{D} \in \bigcap_{e\in\mathcal{E}_{\mathrm{tr}}} \arg\max_{\mathbf{A}\in\mathcal{D}(f)} J_e(\mathbf{A}f). \end{aligned} \tag{17}$$

The same operator must be globally optimal on each source distribution, while the outer encoder remains learnable. This is an exact population constraint: the common operator must attain each environment's maximum, rather than merely give similar coding values. It therefore lets us examine

the limitation of the requirement itself without errors from finite samples or approximate inner optimisation. The operator domain is given by Equation (15).

Theorem 4.1 (A shared optimal coding operator admits the failing encoder)**.** *In model* (6)*, let* $\mathbf{D} = [\mathbf{u}, \mathbf{v}] \in \mathcal{D}(f_S)$ *and* $t = 1 - (\mathbf{u}^\top \mathbf{v})^2$*. Then*

$$J_e(\mathbf{D} f_S) = \frac{1}{2} \log\left(\frac{1 + a + (a^2/4)t}{1 + a + a^2 \delta_e (1 - \delta_e) t} \right). \tag{18}$$

If $\delta_e \neq 1/2$*, its maximisers are exactly the operators with orthogonal unit columns. If* $\delta_e = 1/2$*, every admissible operator is a maximiser. Consequently:*

1. $(f_S, \mathbf{I}_2)$ *satisfies the shared-optimality constraint for any collection of source mismatch probabilities, including distinct ones.*
2. $J^*_{\mathrm{ICR}} = J_\star(a)$*, attained by* $(f_C, \mathbf{I}_2)$*.*
3. *In the positive-noise construction of Theorem 3.3,* $(f_S, \mathbf{I}_2)$ *is a feasible solution whose objective is within* η *of* J^*_{ICR}*, yet its source-optimal classifier has target error at least* $1 - \eta$*.*

The identity operator is also optimal on the reversed target environment.

Why shared optimality does not ensure stable prediction. Theorem 4.1 shows that the same coding operator is optimal for the environmental encoder f_S in every training environment, even when the background–label correlations differ. It also remains optimal after the correlation reverses. However, the classifier still assigns labels according to the training association: when the environmental feature changes from usually matching the label to usually opposing it, these predictions become mostly incorrect. Thus, an operator can remain optimal for the MCR^2 objective even though the relationship between the representation and the label has changed. The shared-optimality constraint does not rule out this failure. With positive noise, the environmental encoder can be arbitrarily close to maximising the constrained training objective while admitting an exactly optimal shared operator and a target prediction error arbitrarily close to one.

4.2 Implications for incorporating invariance

Our analysis takes a first step towards understanding whether the invariance principle can address the OOD limitations of MCR^2. We consider a natural adaptation: requiring the same coding operator to be optimal across training environments. Yet even when this requirement holds exactly and the training objective is arbitrarily close to optimal, prediction can fail on almost every target sample. The reason is that the coding operator can remain optimal even when the relationship between the environmental feature and the label reverses. The constraint favours well-separated representation directions, but does not ensure that those directions retain the same predictive meaning across environments.

It is worth noting that this result does not mean that invariance principle has no role to play in MCR^2. It just shows that simply transferring the shared-optimality requirement widely used in previous work to the MCR^2 objective is not enough. To guarantee reliable OOD generalisation with MCR^2, future approaches need additional new assumptions, learning principles, or regularisation that link representation geometry to stable predictive relationships across environments.

5 Discussion and scope

What is specific to the MCR^2 objective? Distinguishing stable from unstable correlations is a challenge for many learning methods. Our analysis shows precisely how this problem arises for MCR^2: we calculate the global optimum, the environmental encoder's gap from it, and all of its optimal coding operators. In particular, reversing the environmental correlation leaves the entire operator objective unchanged. This explains why a representation can perform well under the MCR^2 objective while supporting unreliable predictions.

The same input support does not mean a small shift. In the identical-support construction, every target input can also occur during training. However, its probability can change substantially. For a source mismatch probability δ and its reversal, the target-to-source density ratio on mismatched inputs is $(1-\delta)/\delta$, which grows without bound as the objective gap approaches zero. Our result therefore does not rule out guarantees that also limit the density ratio or otherwise control the size of the shift. It shows that coding near-optimality alone, with coding precision fixed, cannot guarantee reliable prediction across the full family of shifts considered here.

Near-optimality and the solutions found in practice. With positive source noise, the stable encoder achieves a strictly higher objective than the environmental encoder. Indeed, in our finite model with full source support, every exact global optimiser admits zero source error, and any unrestricted source-optimal classifier also achieves zero target error; see Appendix B.4. Our negative result concerns representations that are arbitrarily close to optimal. Even with exact shared coding optimality, this closeness alone does not guarantee reliable target prediction across the shifts considered here. Whether gradient descent actually selects such representations is a separate question that our analysis does not answer. The exact-optimum failure at zero noise relies on a change in input support.

What additional information could help? Prediction is more stable when each class retains a similar representation distribution across environments. To make this precise, let P and Q have the same label prior p_y. For a fixed encoder f, suppose

$$\mathrm{TV}\left(P(f(\mathbf{X}) \mid Y=y), Q(f(\mathbf{X}) \mid Y=y)\right) \leq \rho_y, \tag{19}$$

where TV denotes total variation distance, a measure of the difference between two probability distributions. Thus, ρ_y bounds how much the representation distribution of class y changes across environments. Every fixed classifier g then satisfies

$$|\mathcal{L}_P(g \circ f) - \mathcal{L}_Q(g \circ f)| \leq \sum_y p_y \rho_y. \tag{20}$$

In other words, small changes in each class's representation distribution ensure a small change in classification error; see Appendix B.6 for the proof. However, equal coding rates alone do not ensure this stability: in our reversal example, the conditional total variation for each class is $|1-2\delta|$, which approaches one as δ approaches zero. Additional assumptions or information, such as paired interventions or known label-preserving transformations, may help establish the stability that the MCR^2 objective alone does not guarantee.

Scope of the model. Our construction uses two classes and two-dimensional unit-norm representations to make the failure mechanism explicit and allow exact analysis. Even in this basic setting, failure occurs despite the availability of an observed stable feature that perfectly predicts the label. This counterexample shows that the MCR^2 objective alone cannot guarantee reliable OOD generalisation. The two-dimensional setting makes this limitation analytically tractable, but does not imply that the underlying problem is confined to two dimensions. Besides, our shared-optimal-coding result shows that directly applying the invariance principle through a common optimal coding operator is insufficient, while leaving open other possible constraints that explicitly address predictive stability.

6 Conclusion

We studied the limits of MCR^2 for OOD generalisation and showed that well-separated representations alone do not ensure reliable prediction across environments. An encoder relying on an environmental feature can achieve the global optimum yet fail on every target sample when the feature–label correlation reverses and input support changes. Even when training and test inputs share the same support, the encoder can be arbitrarily close to optimal while making incorrect predictions on almost every target sample, despite the availability of a perfectly stable feature. Requiring the same coding operator to be optimal across training environments does not resolve this problem: the operator can remain optimal even when the relationship between representations and labels reverses. Finite-sample experiments support these findings. We summarise that for MCR^2 to guarantee reliable OOD generalisation, additional new assumptions or learning principles are needed to ensure that predictive relationships remain stable across environments.

REPRODUCIBILITY STATEMENT

The main text specifies the model, encoder class, and operator domain used in our theoretical analysis. Complete proofs and boundary cases are provided in the appendices, and Figure 4 illustrates the exact construction. Appendix D gives the sample sizes, parameter settings, and repetition counts for the numerical experiments, which evaluate prescribed encoders without training neural networks. Appendix E describes how we train and evaluate the encoder on Waterbirds. Code for reproducing the experiments and generating the figures are included in the supplementary material.

AI USE STATEMENT

AI tools (OpenAI GPT-5 Sol and GPT-6 Astra) were used to polish the writing language, check mathematical proofs, and refine the LaTeX source.

REFERENCES

Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J. Ballard, Joshua Bambrick, et al. Accurate structure prediction of biomolecular interactions with AlphaFold 3. *Nature*, 630:493–500, 2024.

Kartik Ahuja, Karthikeyan Shanmugam, Kush Varshney, and Amit Dhurandhar. Invariant risk minimization games. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 145–155. PMLR, 2020.

Kartik Ahuja, Ethan Caballero, Dinghuai Zhang, Jean-Christophe Gagnon-Audet, Yoshua Bengio, Ioannis Mitliagkas, and Irina Rish. Invariance principle meets information bottleneck for out-of-distribution generalization. In *Advances in Neural Information Processing Systems*, volume 34, pp. 3438–3450, 2021.

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

Ali Atghaei and Mohammad Rahmati. Supervised and semi-supervised domain generalization via consistency learning and maximum coding rate reduction. *Neurocomputing*, 673:132830, 2026.

Christina Baek, Ziyang Wu, Kwan Ho Ryan Chan, Tianjiao Ding, Yi Ma, and Benjamin D. Haeffele. Efficient maximal coding rate reduction by variational forms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 500–508, 2022.

Kwan Ho Ryan Chan, Yaodong Yu, Chong You, Haozhi Qi, John Wright, and Yi Ma. ReduNet: A white-box deep network from the principle of maximizing rate reduction. *Journal of Machine Learning Research*, 23(114):1–103, 2022.

Tianzhe Chu, Shengbang Tong, Tianjiao Ding, Xili Dai, Benjamin D. Haeffele, René Vidal, and Yi Ma. Image clustering via the principle of rate reduction in the age of pre-trained models. In *International Conference on Learning Representations*, 2024.

Xili Dai, Shengbang Tong, Mingyang Li, Ziyang Wu, Michael Psenka, Kwan Ho Ryan Chan, Pengyuan Zhai, Yaodong Yu, Xiaojun Yuan, Heung-Yeung Shum, and Yi Ma. CTRL: Closed-loop transcription to an LDR via minimaxing rate reduction. *Entropy*, 24(4):456, 2022.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, 2019.

Thomas Elsken, Jan Hendrik Metzen, and Frank Hutter. Neural architecture search: A survey. *Journal of Machine Learning Research*, 20(55):1–21, 2019.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *International Conference on Learning Representations*, 2021.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

Matthias Hein and Jean-Yves Audibert. Intrinsic dimensionality estimation of submanifolds in $\mathbb{R}^d$. In *Proceedings of the 22nd international conference on Machine learning*, pp. 289–296, 2005.

Frank Hutter, Lars Kotthoff, and Joaquin Vanschoren (eds.). *Automated Machine Learning: Methods, Systems, Challenges*. Springer, 2019.

Pritish Kamath, Akilesh Tangella, Danica J. Sutherland, and Nathan Srebro. Does invariant risk minimization capture invariance? In *Proceedings of the 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pp. 4069–4077, 2021.

Elia Kaufmann, Leonard Bauersfeld, Antonio Loquercio, Matthias Müller, Vladlen Koltun, and Davide Scaramuzza. Champion-level drone racing using deep reinforcement learning. *Nature*, 620: 982–987, 2023.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton Earnshaw, Imran Haque, Sara Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. WILDS: A benchmark of in-the-wild distribution shifts. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 5637–5664, 2021.

David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (REx). In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 5815–5826. PMLR, 2021.

Zachary C. Lipton. The mythos of model interpretability. *Communications of the ACM*, 61(10): 36–43, 2018.

Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Rätsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 4114–4124, 2019.

Yi Ma, Harm Derksen, Wei Hong, and John Wright. Segmentation of multivariate mixed data via lossy data coding and compression. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(9):1546–1562, 2007.

Yi Ma, Doris Tsao, and Heung-Yeung Shum. On the principles of parsimony and self-consistency for the emergence of intelligence. *Frontiers of Information Technology & Electronic Engineering*, 23 (9):1298–1323, 2022.

Vishal Monga, Yuelong Li, and Yonina C Eldar. Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing. *IEEE Signal Processing Magazine*, 38(2):18–44, 2021.

Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation. In *Proceedings of the 30th International Conference on Machine Learning*, volume 28(1) of *Proceedings of Machine Learning Research*, pp. 10–18, 2013.

Vaishnavh Nagarajan, Anders Andreassen, and Behnam Neyshabur. Understanding the failure modes of out-of-distribution generalization. In *International Conference on Learning Representations*, 2021.

Jonas Peters, Peter Bühlmann, and Nicolai Meinshausen. Causal inference by using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 78(5):947–1012, 2016.

Phillip Pope, Chen Zhu, Ahmed Abdelkader, Micah Goldblum, and Tom Goldstein. The intrinsic dimension of images and its impact on learning. In *International Conference on Learning Representations*, 2021.

Mateo Rojas-Carulla, Bernhard Schölkopf, Richard Turner, and Jonas Peters. Invariant models for causal transfer learning. *Journal of Machine Learning Research*, 19(36):1–34, 2018.

Elan Rosenfeld, Pradeep Ravikumar, and Andrej Risteski. The risks of invariant risk minimization. In *International Conference on Learning Representations*, 2021.

Dominik Rothenhäusler, Nicolai Meinshausen, Peter Bühlmann, and Jonas Peters. Anchor regression: Heterogeneous data meet causality. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 83(2):215–246, 2021.

Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1:206–215, 2019.

Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *International Conference on Learning Representations*, 2020a.

Shiori Sagawa, Aditi Raghunathan, Pang Wei Koh, and Percy Liang. An investigation of why overparameterization exacerbates spurious correlations. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 8346–8356, 2020b.

Bernhard Schölkopf, Dominik Janzing, Jonas Peters, Eleni Sgouritsa, Kun Zhang, and Joris Mooij. On causal and anticausal learning. In *Proceedings of the 29th International Conference on Machine Learning*, pp. 1255–1262, 2012.

Stefano Spigler, Mario Geiger, and Matthieu Wyart. Asymptotic learning curves of kernel methods: Empirical data versus teacher–student paradigm. *Journal of Statistical Mechanics: Theory and Experiment*, 2020(12):124001, 2020.

Shengbang Tong, Xili Dai, Ziyang Wu, Mingyang Li, Brent Yi, and Yi Ma. Incremental learning of structured memory via closed-loop transcription. In *International Conference on Learning Representations*, 2023.

Shengbang Tong, Xili Dai, Yubei Chen, Mingyang Li, Zengyi Li, Brent Yi, Yann LeCun, and Yi Ma. Unsupervised learning of structured representations via closed-loop transcription. In *Conference on Parsimony and Learning*, volume 234 of *Proceedings of Machine Learning Research*, pp. 440–457, 2024.

Peng Wang, Huikang Liu, Druv Pai, Yaodong Yu, Zhihui Zhu, Qing Qu, and Yi Ma. A global geometric analysis of maximal coding rate reduction. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 51012–51040, 2024.

Peng Wang, Yifu Lu, Yaodong Yu, Druv Pai, Qing Qu, and Yi Ma. Attention-only transformers via unrolled subspace denoising. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 63840–63859, 2025.

Olivia Wiles, Sven Gowal, Florian Stimberg, Sylvestre-Alvise Rebuffi, Ira Ktena, Krishnamurthy Dvijotham, and Taylan Cemgil. A fine-grained analysis on distribution shift. In *International Conference on Learning Representations*, 2022.

John Wright and Yi Ma. *High-Dimensional Data Analysis with Low-Dimensional Models: Principles, Computation, and Applications*. Cambridge University Press, 2022.

Ziyang Wu, Christina Baek, Chong You, and Yi Ma. Incremental learning via rate reduction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1125–1133, 2021.

Ziyang Wu, Jingyuan Zhang, Druv Pai, Xudong Wang, Chandan Singh, Jianwei Yang, Jianfeng Gao, and Yi Ma. Simplifying DINO via coding rate regularization. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 68036–68059, 2025.

Jinrui Yang, Xianhang Li, Druv Pai, Yuyin Zhou, Yi Ma, Yaodong Yu, and Cihang Xie. Scaling white-box transformers for vision. In *Advances in Neural Information Processing Systems*, volume 37, pp. 36995–37019, 2024.

Yaodong Yu, Kwan Ho Ryan Chan, Chong You, Chaobing Song, and Yi Ma. Learning diverse and discriminative representations via the principle of maximal coding rate reduction. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9422–9434, 2020.

Yaodong Yu, Sam Buchanan, Druv Pai, Tianzhe Chu, Ziyang Wu, Shengbang Tong, Hao Bai, Yuexiang Zhai, Benjamin D. Haeffele, and Yi Ma. White-box transformers via sparse rate reduction: Compression is all there is? *Journal of Machine Learning Research*, 25(300):1–128, 2024.

Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4):4396–4415, 2023.

Menghui Zhou, Xulong Wang, Tong Liu, Yun Yang, and Po Yang. Integrating visualised automatic temporal relation graph into multi-task learning for Alzheimer's disease progression prediction. *IEEE Transactions on Knowledge and Data Engineering*, 36(10):5206–5220, 2024.

Barret Zoph and Quoc V. Le. Neural architecture search with reinforcement learning. In *International Conference on Learning Representations*, 2017.

APPENDIX OVERVIEW

The appendices provide background, proofs, and experimental details. The outline below indicates where to find each part and how it connects to the main text.

Appendix A: Related work p. 15

Reviews MCR^2 and white-box models, followed by invariance-based approaches to OOD generalisation. It provides background for the motivation in Section 1, the MCR^2 framework in Section 2, and the invariance formulation in Section 4.

Appendix B: Proofs of the main results p. 17

The proofs correspond to the following results and discussions in the main text:

Appendix	Content	Corresponding main-text result
B.1	Global MCR^2 optimum	Lemma 3.1, Section 3
B.2	Environmental encoder's objective and risk	Proposition 3.2, Section 3
B.3	Near-optimality with identical support; noiseless boundary	Theorem 3.3 and Section 3.1
B.4	Exact optimisation under common support	Exact versus near-optimal solutions, Section 5
B.5	Shared optimal coding operator	Theorem 4.1, Section 4
B.6	Prediction-risk bound from class-conditional stability	Equation (20), Section 5

Appendix C: Additional consequences and distinctions p. 24

Expands on Sections 2–5: the information retained by second moments, a limitation of odd encoders, pooling versus averaging source objectives, the distinction between shared classification decisions and shared probability estimates, and the domain of admissible coding operators.

Appendix D: Numerical validation p. 27

Checks the theoretical predictions using finite samples:

- Figure 5: correlation reversal, corresponding to Proposition 3.2 and Section 3.1.
- Figures 6 and 8: near-optimality with identical support and the effect of sample size, corresponding to Theorem 3.3.
- Figure 7: shared operator optimality despite target failure, corresponding to Theorem 4.1.
- Table 2: the exact-optimum and near-optimum failure regimes discussed in Section 3.

Appendix E: Waterbirds experimental setup p. 29

Documents the data splits, preprocessing, encoder training, linear-probe evaluation, and saliency computation underlying the Waterbirds study in Section 2, Table 1, and Figure 3.

A Related work

A.1 Maximal coding rate reduction and white-box models

Coding-rate methods connect the structure of data with the number of bits needed to represent them. Ma et al. (2007) use lossy coding to segment multivariate data by minimising their total coding length. Building on this idea, Yu et al. (2020) introduce MCR^2 as an objective that encourages diversity within each class and separation between classes. Under suitable dimensional and precision conditions, their geometric analysis characterises orthogonal class subspaces and within-class dimensionality. Wang et al. (2024) further study the critical points and optimisation landscape of a regularised coding-rate formulation, providing a global account of its representation geometry. To reduce the computational cost of coding-rate optimisation, Baek et al. (2022) develop variational formulations that avoid repeated evaluation of class-wise log-determinants.

The emphasis on low-dimensional structure is supported by studies of intrinsic dimension (Hein & Audibert, 2005; Pope et al., 2021) and low-dimensional models for high-dimensional data (Wright & Ma, 2022). Related analyses of kernel learning connect learning curves to the smoothness and effective dimension of the data (Spigler et al., 2020). These results help explain why representation geometry matters for learning, but do not establish that the represented features remain predictive under environmental changes. In a different setting, Locatello et al. (2019) show that unsupervised disentanglement requires assumptions about the model and data. Their result concerns the identification of latent factors; ours concerns the stability of label relationships under a supervised coding objective.

The coding-rate perspective also provides a basis for white-box network design, following the broader approach of deriving interpretable networks by unrolling optimisation algorithms (Monga et al., 2021). Chan et al. (2022) derive ReduNet by unfolding the optimisation of coding-rate reduction into network layers. Yu et al. (2024) develop sparse rate reduction and introduce CRATE, a white-box Transformer whose components have explicit roles in compressing and sparsifying representations. Later work improves the scalability of CRATE through CRATE-α (Yang et al., 2024) and derives Attention-only Transformers by unfolding subspace denoising (Wang et al., 2025). Coding-rate principles have also been applied to incremental learning (Wu et al., 2021; Tong et al., 2023), joint discriminative and generative learning (Dai et al., 2022; Tong et al., 2024), image clustering (Chu et al., 2024), and self-supervised learning (Wu et al., 2025).

These studies establish the value of coding-rate principles for learning structured representations and designing interpretable architectures. Our work examines the predictive reliability of these representations when the environment changes. In domain generalisation, Atghaei & Rahmati (2026) combine coding-rate reduction with classification and feature-consistency objectives. Our analysis clarifies what the coding objective itself guarantees, as well as the limits of a direct shared-optimal-coding extension. The architectural developments provide context for the importance of coding-rate methods; our OOD results concern the objective and the specified coding constraint. This distinction helps identify the additional information needed for reliable prediction across environments.

A.2 The invariance principle and OOD generalisation

Domain generalisation studies how to learn from source environments and predict in unseen ones (Zhou et al., 2023). Models can learn environmental cues that predict labels during training but become unreliable in a new environment (Geirhos et al., 2020; Sagawa et al., 2020a). Sagawa et al. (2020b) study how overparameterisation can worsen performance on groups where spurious correlations do not hold. Nagarajan et al. (2021) identify geometric and statistical factors that lead gradient-based learning to exploit such correlations. These studies examine how training produces unstable predictors; our analysis asks whether the coding objective and its shared-optimality constraint exclude them.

Causal approaches connect generalisation to assumptions about which mechanisms remain stable when data distributions change (Schölkopf et al., 2012). Muandet et al. (2013) learn domain-invariant features while preserving predictive information. Rojas-Carulla et al. (2018) study transfer through subsets of predictors whose relationship with the target is invariant across tasks. Anchor regression provides robustness against specified distribution shifts by using observed exogenous

variables (Rothenhäusler et al., 2021). These approaches make explicit the information or assumptions used to support transfer.

The invariance principle seeks predictive relationships that remain valid across environments. Peters et al. (2016) formalise invariant prediction for causal inference, linking stable predictive relationships to causal structure under suitable assumptions. Building on this principle, Arjovsky et al. (2019) propose IRM, which learns a representation for which the same predictor is optimal in every training environment. REx instead encourages similar prediction risks across environments, aiming to improve robustness to shifts beyond those observed during training (Krueger et al., 2021). Related work formulates invariant learning as a game among environments (Ahuja et al., 2020).

Several studies examine the conditions needed for these ideas to support OOD generalisation. Rosenfeld et al. (2021) show that IRM constraints can allow predictors based on environmental features that fail in sufficiently different test environments. Kamath et al. (2021) show that practical IRM formulations can fail to capture the intended invariances, even in simple population settings. Ahuja et al. (2021) identify limitations of invariance alone in classification and show how an information-bottleneck constraint can help under their assumptions. Empirical studies also highlight the importance of evaluation and model selection when comparing domain-generalisation methods (Gulrajani & Lopez-Paz, 2021). WILDS documents naturally occurring distribution shifts across diverse applications (Koh et al., 2021), while Wiles et al. (2022) compare methods across different types of shift and show that their relative performance depends on the setting.

Our work brings this analysis to the coding-rate framework. We study a direct formulation in which the invariance principle translates into a shared coding operator that is optimal across training environments. The same operator can satisfy this requirement even when the relationship between the representation and the label reverses. This result distinguishes agreement on coding geometry from stability of predictive relationships, and identifies a limit of this particular way of introducing invariance into MCR^2.

B PROOFS OF THE MAIN RESULTS

This appendix gives the complete derivations of the coding bound, the environmental encoder's objective and risk, the identical-support construction, and the shared-optimal-coding result. Throughout the main construction, $a = 2/\epsilon^2 > 0$ is fixed. All encoders and classifiers are measurable; measurability is automatic for functions on the finite input domain used in the construction.

B.1 SHARP BINARY CODING BOUND

Proof of Lemma 3.1. Fix an environment and abbreviate $\mathbf{M}_y = \mathbf{M}_{e,y}(f)$ and $\mathbf{M} = (\mathbf{M}_0 + \mathbf{M}_1)/2$. The second identity uses the balanced label prior. We first establish separate bounds on the expansion and compression terms, then identify when both bounds are attained.

Step 1: Properties of the second moments. For every $\mathbf{q} \in \mathbb{R}^2$,

$$\mathbf{q}^\top \mathbf{M}_y \mathbf{q} = \mathbb{E}_e[(\mathbf{q}^\top f(\mathbf{X}))^2 \mid Y = y] \geq 0. \tag{21}$$

Thus each conditional moment is symmetric and positive semidefinite. Linearity of trace and the unit-norm constraint give

$$\operatorname{tr}(\mathbf{M}_y) = \mathbb{E}_e[\operatorname{tr}(f(\mathbf{X})f(\mathbf{X})^\top) \mid Y = y] = \mathbb{E}_e[\|f(\mathbf{X})\|_2^2 \mid Y = y] = 1. \tag{22}$$

The same properties hold for their average $\mathbf{M}$. In particular, all matrices inside the log-determinants are positive definite because $a > 0$, so the objective is finite.

Step 2: Bound on the expansion term. Let λ_1, λ_2 be the eigenvalues of $\mathbf{M}$. They are nonnegative and sum to one. The spectral decomposition and the arithmetic–geometric mean inequality imply

$$\begin{aligned} \det(\mathbf{I}_2 + a\mathbf{M}) &= (1 + a\lambda_1)(1 + a\lambda_2) \\ &\leq \left(\frac{(1 + a\lambda_1) + (1 + a\lambda_2)}{2} \right)^2 \\ &= \left(1 + \frac{a}{2}(\lambda_1 + \lambda_2)\right)^2 = (1 + a/2)^2. \end{aligned} \tag{23}$$

Consequently, $\frac{1}{2}\log\det(\mathbf{I}_2 + a\mathbf{M}) \leq \log(1 + a/2)$. Since $a > 0$, equality holds precisely when $\lambda_1 = \lambda_2 = 1/2$. A symmetric matrix with these eigenvalues is $\mathbf{I}_2/2$, so this equality condition is independent of the choice of eigenbasis.

Step 3: Bound on each compression term. For a two-dimensional matrix, the determinant expansion gives

$$\det(\mathbf{I}_2 + a\mathbf{M}_y) = 1 + a\operatorname{tr}(\mathbf{M}_y) + a^2\det(\mathbf{M}_y) = 1 + a + a^2\det(\mathbf{M}_y) \geq 1 + a. \tag{24}$$

The last inequality follows because both eigenvalues of $\mathbf{M}_y$ are nonnegative. Equality holds if and only if its determinant is zero. Together with trace one, this forces its eigenvalues to be 1 and 0. Equivalently,

$$\mathbf{M}_y = \mathbf{v}_y\mathbf{v}_y^\top, \qquad \|\mathbf{v}_y\|_2 = 1. \tag{25}$$

Thus equality in the conditional bound means that the second moment is a rank-one orthogonal projector.

Step 4: Combine the bounds and characterise equality. With the two class weights equal to $1/2$, the objective is

$$J_e(f) = \frac{1}{2}\log\det(\mathbf{I}_2 + a\mathbf{M}) - \frac{1}{4}\sum_{y=0}^{1}\log\det(\mathbf{I}_2 + a\mathbf{M}_y). \tag{26}$$

Applying the preceding inequalities gives $J_e(f) \leq \log(1 + a/2) - \frac{1}{2}\log(1 + a) = J_\star(a)$. To verify that no slack can cancel, write

$$\begin{aligned} J_\star(a) - J_e(f) = &\left[\log(1 + a/2) - \frac{1}{2}\log\det(\mathbf{I}_2 + a\mathbf{M})\right] \\ &+ \frac{1}{4}\sum_{y=0}^{1}\left[\log\det(\mathbf{I}_2 + a\mathbf{M}_y) - \log(1 + a)\right]. \end{aligned} \tag{27}$$

Every bracket is nonnegative. The total is zero if and only if every bracket is zero. Hence equality requires $\mathbf{M} = \mathbf{I}_2/2$ and $\mathbf{M}_y = \mathbf{v}_y\mathbf{v}_y^\top$ for both labels. These conditions imply

$$\mathbf{v}_0\mathbf{v}_0^\top + \mathbf{v}_1\mathbf{v}_1^\top = \mathbf{I}_2. \tag{28}$$

Taking the quadratic form along $\mathbf{v}_0$ yields

$$1 + (\mathbf{v}_0^\top\mathbf{v}_1)^2 = 1, \tag{29}$$

so $\mathbf{v}_0^\top\mathbf{v}_1 = 0$. Conversely, two orthogonal unit vectors form an orthonormal basis of $\mathbb{R}^2$. Their projectors sum to $\mathbf{I}_2$, and both the expansion and compression bounds are attained. This proves the stated necessary and sufficient equality condition.

Step 5: Attainment in every training environment. In model (6), $C = Y$ almost surely, so $f_C(\mathbf{X}) = \mathbf{b}_Y$. Its conditional moments are $\mathbf{b}_y\mathbf{b}_y^\top$ in every environment, regardless of δ_e. Therefore $J_e(f_C) = J_\star(a)$ for every e. For any other $f \in \mathcal{F}$,

$$J_{\mathrm{tr}}(f) = \sum_e w_e J_e(f) \le \sum_e w_e J_\star(a) = J_\star(a). \tag{30}$$

The encoder f_C belongs to $\mathcal{F}$ and attains this bound. Thus the bound is the global maximum over the full encoder class, rather than only over the two encoders displayed in the construction. □

B.2 Exact coding value and source-optimal head

Proof of Proposition 3.2. Fix an environment with mismatch probability $\delta \in [0, 1]$. We calculate the complete distribution on the two output codes before evaluating the objective and prediction risk.

Step 1: Conditional and marginal code distributions. Since $S = Y \oplus B_e$ and B_e is independent of Y,

$$P_e(S = s \mid Y = y) = \begin{cases} 1-\delta, & s = y, \\ \delta, & s \neq y. \end{cases} \tag{31}$$

The balanced label prior implies, for each $s \in \{0, 1\}$,

$$P_e(S = s) = \frac{1}{2}(1-\delta) + \frac{1}{2}\delta = \frac{1}{2}. \tag{32}$$

Because $f_S(\mathbf{X}) = \mathbf{b}_S$, the representation marginal is therefore uniform on the two codes for every value of δ, including the endpoints. The conditional and global moments are

$$\mathbf{M}_0 = \begin{pmatrix} 1-\delta & 0 \\ 0 & \delta \end{pmatrix}, \qquad \mathbf{M}_1 = \begin{pmatrix} \delta & 0 \\ 0 & 1-\delta \end{pmatrix}, \qquad \mathbf{M} = \frac{1}{2}\mathbf{I}_2. \tag{33}$$

Step 2: Evaluate the log-determinants. The global determinant is $(1 + a/2)^2$. Each conditional determinant is

$$\begin{aligned} (1 + a\delta)(1 + a(1-\delta)) &= 1 + a\delta + a(1-\delta) + a^2\delta(1-\delta) \\ &= 1 + a + a^2\delta(1-\delta). \end{aligned} \tag{34}$$

Substituting into the population objective and combining the two identical conditional terms gives

$$\begin{aligned} J_\delta(f_S) &= \log(1 + a/2) - \frac{1}{2}\log\left(1 + a + a^2\delta(1-\delta)\right) \\ &= \log(1 + a/2) - \frac{1}{2}\log(1 + a) - \frac{1}{2}\log\left(1 + \frac{a^2}{1+a}\delta(1-\delta)\right) \\ &= J_\star(a) - \Gamma_a(\delta). \end{aligned} \tag{35}$$

Here we factored $1 + a$ from the conditional determinant; the factor is strictly positive. The formula holds on the entire interval $[0, 1]$. In particular, the gap vanishes at $\delta = 0$ and $\delta = 1$, is strictly positive for $0 < \delta < 1$, and is unchanged by replacing δ with $1 - \delta$.

Step 3: Risk of the fixed classifier. The classifier g_0 reads the code index, so $g_0(f_S(\mathbf{X})) = S$. Its error event is exactly $\{B_e = 1\}$, giving

$$\mathcal{L}_e(g_0 \circ f_S) = P_e(S \neq Y) = P_e(B_e = 1) = \delta_e. \tag{36}$$

For the stable encoder, $g_0(f_C(\mathbf{X})) = C = Y$ almost surely. Its risk is zero in every source and target environment.

Step 4: Optimality on the weighted source mixture. Let $P_{\mathrm{tr}} = \sum_e w_e P_e$ and $\bar{\delta} = \sum_e w_e \delta_e$. Minimising the weighted source risk is equivalent to minimising risk under this mixture. For either code index s,

$$\begin{aligned} P_{\mathrm{tr}}(Y = s, S = s) &= \frac{1}{2}\sum_e w_e(1 - \delta_e) = \frac{1}{2}(1 - \bar{\delta}), \\ P_{\mathrm{tr}}(Y = 1 - s, S = s) &= \frac{1}{2}\sum_e w_e \delta_e = \frac{1}{2}\bar{\delta}. \end{aligned} \tag{37}$$

As $P_{\mathrm{tr}}(S = s) = 1/2$, Bayes' rule gives

$$P_{\mathrm{tr}}(Y = s \mid S = s) = 1 - \bar{\delta}, \qquad P_{\mathrm{tr}}(Y = 1 - s \mid S = s) = \bar{\delta}. \tag{38}$$

Predicting s at $\mathbf{b}_s$ has conditional error $\bar{\delta}$, whereas predicting $1 - s$ has conditional error $1 - \bar{\delta}$. If $\bar{\delta} < 1/2$, the former is strictly smaller at both codes. More explicitly, for any deterministic binary classifier g, let $m(g)$ be the number of codes on which $g(\mathbf{b}_s) \neq s$. Then

$$\mathcal{L}_{\mathrm{tr}}(g \circ f_S) = \bar{\delta} + \frac{1 - 2\bar{\delta}}{2} m(g). \tag{39}$$

This is uniquely minimised on the representation support by $m(g) = 0$, which is exactly g_0. Allowing randomised decisions does not lower the minimum: their conditional errors are convex combinations of the two errors just calculated. Values of g outside the two-code support do not affect its risk. Finally, $g_0(\mathbf{z}) = \mathbf{1}\{z_2 > z_1\}$ implements the optimal assignments on the basis codes, so a linear decision boundary attains the unrestricted optimum. □

B.3 Near-optimality with identical input support

Proof of Theorem 3.3. We first establish the quantitative gap estimate for arbitrary small source mismatch probabilities. We then give an explicit choice of environments and verify all requirements of the theorem.

Step 1: Bound the coding gap. For $x \geq 0$, $\log(1 + x) \leq x$. Applying this with $x = a^2\delta(1 - \delta)/(1 + a)$ yields

$$0 \leq \Gamma_a(\delta) \leq \frac{a^2}{2(1 + a)}\delta(1 - \delta) \leq \frac{a^2}{2(1 + a)}\delta, \qquad 0 \leq \delta \leq 1. \tag{40}$$

By Lemma 3.1 and Proposition 3.2,

$$\begin{aligned} \max_{f \in \mathcal{F}} J_{\mathrm{tr}}(f) - J_{\mathrm{tr}}(f_S) &= J_\star(a) - \sum_e w_e[J_\star(a) - \Gamma_a(\delta_e)] \\ &= \sum_e w_e \Gamma_a(\delta_e) \\ &\leq \frac{a^2}{2(1 + a)} \sum_e w_e \delta_e \\ &\leq \frac{a^2}{2(1 + a)} \delta_{\max}. \end{aligned} \tag{41}$$

The final step uses positive weights summing to one and $\delta_e \leq \delta_{\max}$. This proves Equation (13) without requiring equal source weights.

Step 2: Choose the source and target environments. Fix the prescribed $a > 0$, finite positive integer E, source weights, and $\eta > 0$. Define

$$r = \min\left\{\frac{1}{4}, \frac{\eta}{2}, \frac{(1+a)\eta}{a^2}\right\} > 0, \qquad \delta_e = \frac{e}{E}r \quad (e = 1, \ldots, E), \qquad \delta_{\mathrm{te}} = 1 - r. \tag{42}$$

All three terms defining r are strictly positive. Hence $0 < r \leq 1/4$, and the source parameters satisfy

$$0 < \delta_1 < \cdots < \delta_E = r < \frac{1}{2}. \tag{43}$$

For $E = 1$, this simply means $\delta_1 = r$. The target parameter lies in $[3/4, 1)$, so its environmental feature is more often incorrect than correct. The largest source mismatch probability is $\delta_{\max} = r$.

Step 3: Verify objective quality and source optimality. The gap estimate and $r \leq (1+a)\eta/a^2$ give

$$0 \leq \max_{f \in \mathcal{F}} J_{\mathrm{tr}}(f) - J_{\mathrm{tr}}(f_S) \leq \frac{a^2 r}{2(1+a)} \leq \frac{\eta}{2} \leq \eta. \tag{44}$$

Also, $\bar{\delta} = \sum_e w_e \delta_e \leq r < 1/2$. Proposition 3.2 therefore establishes that g_0 is source-optimal on f_S. No target labels are used to select this classifier.

Step 4: Verify target failure and the stable alternative. Keeping the same classifier at test time gives

$$\mathcal{L}_{\mathrm{te}}(g_0 \circ f_S) = \delta_{\mathrm{te}} = 1 - r \geq 1 - \frac{\eta}{2} \geq 1 - \eta. \tag{45}$$

At the same time, $\mathcal{L}_{\mathrm{te}}(g_0 \circ f_C) = 0$. The failure thus occurs despite the availability of an encoder and classifier that are perfectly accurate in every environment.

Step 5: Verify common support. For an environment with parameter δ, the input probabilities are

(c, s)	$(0,0)$	$(0,1)$	$(1,0)$	$(1,1)$
$P_e(C = c, S = s)$	$(1-\delta)/2$	$\delta/2$	$\delta/2$	$(1-\delta)/2.$

(46)

Each source has $0 < \delta_e < 1$ and the target has $0 < 1 - r < 1$. Every entry is therefore strictly positive in every environment, and every input support equals $\{0,1\}^2$. Since $Y = C$ deterministically, the joint support of $(\mathbf{X}, Y)$ is also identical: it consists of the four triples $((c,s),c)$. The theorem concerns a change in probabilities and predictive relationships on this shared support, rather than the appearance of previously impossible inputs.

These checks establish the two inequalities, source optimality, the stable zero-risk alternative, and common support for any specified finite number of sources and any specified positive weights. The encoder f_S itself is the same throughout; only the environment parameters are chosen as a function of the requested tolerance. □

The exact-optimum boundary. If every source has $\delta_e = 0$, then $f_C = f_S$ almost surely on every source, and both attain $J_\star(a)$. A target with $\delta_{\mathrm{te}} = 1$ has $S = 1 - Y$, so $g_0 \circ f_S$ is wrong with probability one. The source input support is $\{(0,0),(1,1)\}$, while the target support is $\{(0,1),(1,0)\}$. This is why the exact boundary example cannot be used as a proof of exact-optimum failure with identical input support. In the positive-noise construction, $\Gamma_a(\delta_e) > 0$ for every source, so f_C is strictly better in objective value than f_S. The theorem establishes the insufficiency of a small objective gap, not inevitable failure of an exact optimiser.

Why precision is fixed in the theorem. The coefficient $a^2/[2(1+a)]$ depends on coding precision. For a fixed $\delta \in (0,1)$, the exact gap is $\frac{1}{2}\log(1 + a^2\delta(1-\delta)/(1+a))$, which grows without bound as $a \to \infty$. The theorem instead fixes any $a > 0$ and then selects sufficiently small source mismatch probabilities. Its quantifier order is therefore important: it does not assert uniform near-optimality in precision at a fixed noise level.

B.4 EXACT OPTIMISATION UNDER COMMON SUPPORT

The positive-noise result concerns representations arbitrarily close to the optimum. The following consequence of Lemma 3.1 makes the distinction from exact optimisation explicit.

Proposition B.1 (Exact optimisers in the full-support model)**.** *In model (6), suppose $0 < \delta_e < 1$ in every training environment and all source weights are positive. If $f \in \mathcal{F}$ attains $J_{\mathrm{tr}}(f) = J_\star(a)$, then every source-optimal classifier over all functions on the representation has zero risk in every target environment from the same model, for any $\delta_{\mathrm{te}} \in [0, 1]$. The same statement holds for the representation $\mathbf{D}f$ of any feasible pair attaining J^*_{ICR}.*

Proof. We use the equality condition of the coding bound to identify which representations an exact optimiser can assign to each input.

Step 1: Equality holds in each source. By Lemma 3.1, $J_\star(a) - J_e(f) \geq 0$ for every e. Since all $w_e > 0$ and

$$0 = J_\star(a) - J_{\mathrm{tr}}(f) = \sum_e w_e [J_\star(a) - J_e(f)], \tag{47}$$

each term must be zero. Choose any source environment. The equality characterisation gives unit vectors $\mathbf{v}_0, \mathbf{v}_1$ such that $\mathbf{v}_0 \perp \mathbf{v}_1$ and $\mathbf{M}_{e,y}(f) = \mathbf{v}_y \mathbf{v}_y^\top$.

Step 2: Every input of a class maps to its class line. Fix $y \in \{0, 1\}$ and a unit vector $\mathbf{q}_y \perp \mathbf{v}_y$. Since $C = Y$, the conditional moment identity implies

$$0 = \mathbf{q}_y^\top \mathbf{M}_{e,y}(f) \mathbf{q}_y = \sum_{s=0}^{1} P_e(S = s \mid Y = y) \big(\mathbf{q}_y^\top f(y, s)\big)^2. \tag{48}$$

Both conditional probabilities are strictly positive because $0 < \delta_e < 1$. Each summand is nonnegative, so both squares vanish. Thus $f(y, s)$ lies on the line spanned by $\mathbf{v}_y$. The unit-norm condition further gives

$$f(y, s) \in \{\mathbf{v}_y, -\mathbf{v}_y\}, \qquad s \in \{0, 1\}. \tag{49}$$

The two class lines are orthogonal, so their unit-vector sets are disjoint. The encoder may still use s to choose a sign, but that choice cannot move an input onto the other class line.

Step 3: Source-optimal predictions are correct on all possible inputs. A classifier assigning label y to each attained code in $\{\mathbf{v}_y, -\mathbf{v}_y\}$ has zero source risk. Hence the optimal source risk over all classifiers is zero. Every input in $\{0, 1\}^2$ has positive source probability, so any source-optimal classifier must classify the representation of every one of these inputs correctly. Otherwise its source risk would be strictly positive.

A target environment in the stated family changes only the probabilities assigned to these inputs; it retains the deterministic label $Y = C$. The same classifier is therefore correct on every target input, including when $\delta_{\mathrm{te}} = 0$ or 1 reduces the target support. Its target risk is zero.

Step 4: Exact constrained optima. Theorem 4.1 establishes $J^*_{\mathrm{ICR}} = J_\star(a)$. For a feasible maximising pair, let $h(\mathbf{x}) = \mathbf{D}f(\mathbf{x})$. Admissibility ensures $h \in \mathcal{F}$, and attainment gives $\sum_e w_e J_e(h) = J_\star(a)$. Applying the preceding steps to h proves the final assertion. □

This proposition concerns an unrestricted source-optimal classifier. A linear classifier need not separate two classes that each occupy both signs of their respective lines. For the particular encoders f_C and f_S, however, the linear rule in Section 2 attains the source optimum as stated. The proposition also uses the finite model and full source support; it does not extend the zero-risk conclusion to arbitrary input spaces or label mechanisms.

B.5 SHARED CODING OPTIMALITY

Proof of Theorem 4.1. We characterise the entire inner operator optimisation, then verify feasibility and objective quality in the outer problem.

Step 1: Characterise the admissible operators. Both $\mathbf{b}_0$ and $\mathbf{b}_1$ occur in the image of f_S on the fixed input domain $\mathcal{X}$. For a linear operator $\mathbf{D} = [\mathbf{u}, \mathbf{v}]$, its outputs on these codes are $\mathbf{D}\mathbf{b}_0 = \mathbf{u}$ and $\mathbf{D}\mathbf{b}_1 = \mathbf{v}$. Thus the requirement $\|\mathbf{D}f_S(\mathbf{x})\|_2 = 1$ for all $\mathbf{x} \in \mathcal{X}$ is equivalent to $\|\mathbf{u}\|_2 = \|\mathbf{v}\|_2 = 1$. The same argument applies to f_C. This uses the full input domain specified in the operator constraint, including when an individual environment has a smaller support.

Write $\gamma = \mathbf{u}^\top \mathbf{v}$ and $t = 1 - \gamma^2$. Cauchy–Schwarz gives $0 \le t \le 1$. Every value in this interval is attainable: choosing $\mathbf{u} = (1, 0)^\top$ and $\mathbf{v} = (\sqrt{1-t}, \sqrt{t})^\top$ produces unit columns with the required value of t. In particular, $t = 0$ allows parallel or antiparallel columns, and $t = 1$ means orthogonal columns.

Step 2: Compute the moments after applying the operator. For $p \in [0, 1]$, define

$$\mathbf{A}_p = p\mathbf{u}\mathbf{u}^\top + (1-p)\mathbf{v}\mathbf{v}^\top = \mathbf{D}\,\mathrm{diag}(p, 1-p)\mathbf{D}^\top. \tag{50}$$

It is positive semidefinite and has trace $p\|\mathbf{u}\|_2^2 + (1-p)\|\mathbf{v}\|_2^2 = 1$. Its determinant is

$$\begin{aligned}\det(\mathbf{A}_p) &= p(1-p)(\det \mathbf{D})^2 \\ &= p(1-p)\det(\mathbf{D}^\top \mathbf{D}) \\ &= p(1-p)\det\begin{pmatrix} 1 & \gamma \\ \gamma & 1 \end{pmatrix} = p(1-p)t.\end{aligned} \tag{51}$$

No inverse of $\mathbf{D}$ is used, so these identities remain valid for singular operators and for $p = 0$ or $p = 1$. The two-dimensional determinant identity now gives

$$\det(\mathbf{I}_2 + a\mathbf{A}_p) = 1 + a + a^2 p(1-p)t. \tag{52}$$

The marginal code probabilities are $1/2, 1/2$, so the global moment of $\mathbf{D}f_S$ is $\mathbf{A}_{1/2}$. Its class-conditional moments are $\mathbf{A}_{1-\delta_e}$ and $\mathbf{A}_{\delta_e}$. Their determinants are equal, although the matrices need not be equal.

Step 3: Evaluate the full operator objective. Substituting these moments into the population objective gives

$$\begin{aligned}J_e(\mathbf{D}f_S) &= \frac{1}{2}\log(1 + a + a^2 t/4) - \frac{1}{2}\log(1 + a + a^2\delta_e(1-\delta_e)t) \\ &= \frac{1}{2}\log\left(\frac{1 + a + (a^2/4)t}{1 + a + a^2\delta_e(1-\delta_e)t}\right).\end{aligned} \tag{53}$$

This proves Equation (18). In particular, all dependence on the two unit columns is through t. All logarithm arguments are positive, so differentiation is valid throughout $[0, 1]$, with one-sided derivatives at its endpoints.

Step 4: Identify every maximising operator. Let $A = 1 + a$, $B = a^2/4$, and $C = a^2\delta_e(1-\delta_e)$. Here these are scalar constants. Direct differentiation gives

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\left[\frac{1}{2}\log\frac{A + Bt}{A + Ct}\right] &= \frac{1}{2}\left(\frac{B}{A + Bt} - \frac{C}{A + Ct}\right) \\ &= \frac{B(A + Ct) - C(A + Bt)}{2(A + Bt)(A + Ct)} \\ &= \frac{A(B - C)}{2(A + Bt)(A + Ct)}.\end{aligned} \tag{54}$$

Substituting these constants gives

$$\frac{\partial J_e(\mathbf{D}f_S)}{\partial t} = \frac{a^2(1+a)\big(1/4 - \delta_e(1-\delta_e)\big)}{2\big(1 + a + a^2 t/4\big)\big(1 + a + a^2\delta_e(1-\delta_e)t\big)}. \tag{55}$$

The denominator is strictly positive, and

$$\frac{1}{4} - \delta_e(1-\delta_e) = \left(\delta_e - \frac{1}{2}\right)^2. \tag{56}$$

If $\delta_e \neq 1/2$, the objective is strictly increasing in t. Because $t = 1$ is attainable, its maximising operators are exactly those with orthogonal unit columns. If $\delta_e = 1/2$, numerator and denominator inside the logarithm are identical, and $J_e(\mathbf{D}f_S) = 0$ for every admissible operator. This proves both cases in the theorem, including the complete characterisation of the maximiser set.

Step 5: Verify common optimality, including under reversal. The identity $\mathbf{I}_2$ has orthogonal unit columns. It is therefore optimal in every environment with $\delta_e \neq 1/2$ and is also a maximiser when $\delta_e = 1/2$. Consequently, it belongs to the intersection of all source argmax sets, independently of the number or distinctness of their mismatch probabilities. If at least one source has $\delta_e \neq 1/2$, that intersection is exactly the set of orthogonal operators; if all have $\delta_e = 1/2$, it is the entire admissible operator set.

Moreover, replacing δ_e by $1-\delta_e$ leaves $\delta_e(1-\delta_e)$ unchanged. It therefore preserves the full objective function of $\mathbf{D}$, not merely its maximum value. In particular, $\mathbf{I}_2$ remains optimal on the reversed target distribution even though the source classifier's predictions become mostly incorrect there.

Step 6: Determine the outer constrained optimum. For the stable encoder, conditional on $Y = 0$ the output after applying $\mathbf{D}$ is $\mathbf{u}$, and conditional on $Y = 1$ it is $\mathbf{v}$. The preceding computation with $\delta = 0$ gives

$$J_e(\mathbf{D}f_C) = \frac{1}{2}\log\frac{1 + a + a^2t/4}{1 + a}. \tag{57}$$

This function is strictly increasing in t and does not depend on the environment. Hence $(f_C, \mathbf{I}_2)$ is feasible in Equation (17). At $t = 1$, the numerator equals $(1 + a/2)^2$, giving $J_e(f_C) = J_\star(a)$ in every source.

Conversely, for every admissible pair $(f, \mathbf{D})$, the composition $h(\mathbf{x}) = \mathbf{D}f(\mathbf{x})$ is a unit-norm encoder into $\mathbb{R}^2$ on the whole input domain. Lemma 3.1 applies to h in each balanced environment. Thus every feasible pair satisfies

$$\sum_e w_e J_e(\mathbf{D}f) \leq \sum_e w_e J_\star(a) = J_\star(a). \tag{58}$$

Since the feasible pair $(f_C, \mathbf{I}_2)$ attains this bound, the constrained supremum is attained and $J^*_{\mathrm{ICR}} = J_\star(a)$.

Step 7: Verify the near-optimal failing feasible solution. Apply the positive-noise environments constructed in Theorem 3.3. Step 5 establishes exact shared optimality of $\mathbf{I}_2$ for f_S, so $(f_S, \mathbf{I}_2)$ is feasible. Its gap from the constrained optimum is

$$J^*_{\mathrm{ICR}} - \sum_e w_e J_e(\mathbf{I}_2 f_S) = J_\star(a) - J_{\mathrm{tr}}(f_S) \leq \eta. \tag{59}$$

Applying the identity does not change the representation seen by the classifier. The source-optimal rule remains g_0, and its target error remains at least $1 - \eta$. All support and stable-feature properties are inherited from the same construction. This proves the three assertions and the target-optimality statement. □

Why the operator comparison is nontrivial. For example, $\mathbf{D} = [\mathbf{b}_0, \mathbf{b}_0]$ is admissible on the two-code support but maps both codes to the same vector. It has $t = 0$ and coding-rate reduction zero. The identity operator has $t = 1$ and strictly positive rate reduction whenever $\delta \neq 1/2$. The shared-maximiser statement therefore compares genuinely different coding geometries. It is stronger than saying that all orthogonal coordinate changes preserve an objective, but it remains a statement about the explicit domain $\mathcal{D}(f)$.

B.6 A SUFFICIENT DISTRIBUTIONAL CONDITION FOR FIXED-HEAD STABILITY

Proof of Equation (20). Let P and Q have the same label prior p_y. For a fixed encoder f, denote the conditional representation laws by $\mu_y = P(f(\mathbf{X}) \in \cdot \mid Y = y)$ and $\nu_y = Q(f(\mathbf{X}) \in \cdot \mid Y = y)$. Labels of zero prior can be omitted. We use the convention

$$\mathrm{TV}(\mu, \nu) = \sup_A |\mu(A) - \nu(A)|, \tag{60}$$

where the supremum ranges over measurable sets. For a fixed measurable classifier g, define the conditional error set $A_y = \{\mathbf{z} : g(\mathbf{z}) \neq y\}$. The law of total probability gives

$$\mathcal{L}_P(g \circ f) = \sum_y p_y \mu_y(A_y), \qquad \mathcal{L}_Q(g \circ f) = \sum_y p_y \nu_y(A_y). \tag{61}$$

Subtracting these expressions, applying the triangle inequality, and then using the definition of total variation gives

$$\begin{aligned} |\mathcal{L}_P(g \circ f) - \mathcal{L}_Q(g \circ f)| &= \left| \sum_y p_y [\mu_y(A_y) - \nu_y(A_y)] \right| \\ &\leq \sum_y p_y |\mu_y(A_y) - \nu_y(A_y)| \\ &\leq \sum_y p_y \, \mathrm{TV}(\mu_y, \nu_y) \\ &\leq \sum_y p_y \rho_y. \end{aligned} \tag{62}$$

The last step is exactly the class-conditioned stability assumption in Equation (19). The bound applies to any fixed classifier, without requiring source optimality. Equal label priors allow the same weights to appear in the two risk decompositions; matching a scalar coding value is not used in this derivation.

To check sharpness in the reversal construction, the two conditional measures for either class have masses $(1-\delta, \delta)$ and $(\delta, 1-\delta)$ on the two basis codes, with their order depending on the class. Total variation on this two-point support is

$$\frac{1}{2}\left(|(1-\delta) - \delta| + |\delta - (1-\delta)|\right) = |1 - 2\delta|. \tag{63}$$

The fixed classifier g_0 has risks δ and $1-\delta$, whose absolute difference is also $|1-2\delta|$. With class weights $1/2, 1/2$, the right-hand side of the bound is exactly the same quantity. Thus the bound is attained in this example, despite identical coding values before and after reversal. □

C Additional consequences and distinctions

C.1 The second-moment information retained by the objective

Proposition C.1 (Moment equivalence)**.** *Fix the representation dimension, precision, and label prior in an environment. If two encoders f, h have $\mathbf{M}_{e,y}(f) = \mathbf{M}_{e,y}(h)$ for every label, then $J_e(f) = J_e(h)$. If the equalities hold in every source environment, the two weighted source objectives are equal.*

Proof. Write the common class moments as $\mathbf{N}_y = \mathbf{M}_{e,y}(f) = \mathbf{M}_{e,y}(h)$. Since both encoders are evaluated on the same environment, their label priors are the same. Equation (3) therefore gives

$$\mathbf{M}_e(f) = \sum_y p_y^e \mathbf{N}_y = \mathbf{M}_e(h). \tag{64}$$

The representation dimension and precision are fixed, so the same identity matrix and scalar a are used in both objectives. The global log-determinants coincide by the displayed identity, and each conditional log-determinant coincides by the assumed equality of the corresponding class moments. Thus

$$\begin{aligned} J_e(f) - J_e(h) = \frac{1}{2}\left[\log\det(\mathbf{I}_d + a\mathbf{M}_e(f)) - \log\det(\mathbf{I}_d + a\mathbf{M}_e(h))\right] \\ - \frac{1}{2}\sum_y p_y^e \left[\log\det(\mathbf{I}_d + a\mathbf{M}_{e,y}(f)) - \log\det(\mathbf{I}_d + a\mathbf{M}_{e,y}(h))\right] = 0. \end{aligned} \tag{65}$$

If the assumed class-moment equalities hold separately in every source, the same argument gives $J_e(f) = J_e(h)$ for every source. Multiplication by the common source weights and summation then yield $J_{\mathrm{tr}}(f) = J_{\mathrm{tr}}(h)$. No equality of the full conditional representation distributions is needed for this conclusion. □

This observation is an algebraic property, not by itself an OOD impossibility theorem. Equal second moments do not specify the complete class-conditioned distributions. Conversely, the main reversal construction does not have equal *class-indexed* moments across source and target: it swaps them. Its equal coding values follow from that symmetry rather than from Proposition C.1.

C.2 Odd encoders and reflected classes

Proposition C.2 (A restricted encoder-class blind spot)**.** *Suppose binary classes satisfy* $\mathbf{X} \mid Y = -1 \overset{d}{=} -\mathbf{X} \mid Y = +1$. *If an encoder obeys* $f(-\mathbf{x}) = -f(\mathbf{x})$ *on the relevant support and has finite second moments, then its two conditional second moments are equal and its population coding-rate reduction is zero, for any class prior and any positive coding precision.*

Proof. Let $\mathbf{H}(\mathbf{x}) = f(\mathbf{x})f(\mathbf{x})^\top$. Oddness of f makes this matrix-valued function even:

$$\mathbf{H}(-\mathbf{x}) = (-f(\mathbf{x}))(-f(\mathbf{x}))^\top = f(\mathbf{x})f(\mathbf{x})^\top = \mathbf{H}(\mathbf{x}). \tag{66}$$

Finite second moments ensure that its entries are integrable. The assumed reflection identity of the conditional input laws therefore implies

$$\begin{aligned}\mathbf{M}_{-1} &= \mathbb{E}[f(\mathbf{X})f(\mathbf{X})^\top \mid Y = -1] \\ &= \mathbb{E}[f(-\mathbf{X})f(-\mathbf{X})^\top \mid Y = +1] \\ &= \mathbb{E}[f(\mathbf{X})f(\mathbf{X})^\top \mid Y = +1] = \mathbf{M}_{+1}.\end{aligned} \tag{67}$$

Denote this common matrix by $\mathbf{N}$ and the two label probabilities by p_{-1} and p_{+1}. Their sum is one, so the global moment is

$$\mathbf{M} = p_{-1}\mathbf{N} + p_{+1}\mathbf{N} = \mathbf{N}. \tag{68}$$

For any $a > 0$, $\mathbf{I}_d + a\mathbf{N}$ is positive definite. Substitution into the objective gives

$$J(f) = \frac{1}{2}\log\det(\mathbf{I}_d + a\mathbf{N}) - \frac{1}{2}(p_{-1} + p_{+1})\log\det(\mathbf{I}_d + a\mathbf{N}) = 0. \tag{69}$$

Thus the cancellation does not require balanced classes. For nondegenerate priors both conditional laws are used; if a class has zero prior, the objective is also zero because its single positive-weight conditional moment equals the global moment. □

For a concrete example, let Y be uniform on $\{-1, +1\}$ and $\mathbf{X} \mid Y = y \sim \mathcal{N}(y\boldsymbol{\mu}, \sigma^2\mathbf{I}_d)$, with $\boldsymbol{\mu} \neq 0$ and $\sigma > 0$. The encoder $f(\mathbf{X}) = \mathbf{X}/\|\mathbf{X}\|_2$ is well-defined almost surely and odd. Its coding-rate reduction is zero. Nevertheless, the rule $g(\mathbf{z}) = \operatorname{sign}(\boldsymbol{\mu}^\top\mathbf{z})$ has error

$$P(Y\boldsymbol{\mu}^\top\mathbf{X} < 0) = \Phi(-\|\boldsymbol{\mu}\|_2/\sigma), \tag{70}$$

where Φ is the standard normal cumulative distribution function. To see this directly, write $\mathbf{X} = Y\boldsymbol{\mu} + \sigma\boldsymbol{\xi}$ with $\boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$ independent of Y. Multiplication by the positive scalar $1/\|\mathbf{X}\|_2$ does not change the sign of the classifier score. Moreover,

$$Y\boldsymbol{\mu}^\top\mathbf{X} = \|\boldsymbol{\mu}\|_2^2 + \sigma Y\boldsymbol{\mu}^\top\boldsymbol{\xi} \sim \mathcal{N}(\|\boldsymbol{\mu}\|_2^2, \sigma^2\|\boldsymbol{\mu}\|_2^2). \tag{71}$$

The distribution is continuous, so the zero-score event has probability zero. Standardising the score gives the displayed error formula, which approaches zero as $\|\boldsymbol{\mu}\|_2/\sigma \to \infty$. The example shows that predicting a label from a representation and distinguishing its classes through uncentred second moments are different properties. It does not show that a general nonlinear encoder is unable to learn a representation with positive coding-rate reduction. Non-odd feature maps can change these moments. This separate observation is therefore not used as a proof of the main OOD result.

C.3 Pooling sources versus averaging their objectives

The main text uses $\sum_e w_e J_e(f)$. A second possible training objective is $J_{\text{pool}}(f)$, computed after pooling sources with weights w_e. These are generally different because $\log\det$ is nonlinear. For the environmental encoder in the present model, pooling simply replaces δ_e with $\bar{\delta} = \sum_e w_e\delta_e$. Therefore

$$J_{\text{pool}}(f_S) = J_\star(a) - \Gamma_a(\bar{\delta}), \qquad J_{\text{pool}}(f_C) = J_\star(a). \tag{72}$$

As $\bar{\delta} \leq \delta_{\max}$, the bound in Equation (13) also bounds the pooled gap. The identical-support near-optimality construction thus applies to either objective, but their numerical values should not be identified for general encoders.

C.4 Source probability estimates and IRM

For f_S in environment e, the Bayes conditional probability is

$$P_e(Y = 1 \mid f_S(\mathbf{X}) = \mathbf{b}_0) = \delta_e, \qquad P_e(Y = 1 \mid f_S(\mathbf{X}) = \mathbf{b}_1) = 1 - \delta_e. \tag{73}$$

When every $\delta_e < 1/2$, these probabilities induce the same optimal 0–1 rule. They are not equal as probability functions when the source mismatch probabilities differ. For $0 < \delta_e < 1$, the logistic-optimal logits are

$$\log \frac{\delta_e}{1 - \delta_e} \quad \text{at } \mathbf{b}_0, \qquad \log \frac{1 - \delta_e}{\delta_e} \quad \text{at } \mathbf{b}_1. \tag{74}$$

At $\delta_e = 0$ or 1, the corresponding logistic loss approaches its infimum only as the logits diverge; there is no finite optimal logit. Thus the environmental encoder in the distinct-source construction does not automatically satisfy a common logistic-optimal classifier condition. Agreement of decision boundaries should not be substituted for the appropriate loss-specific optimality requirement. In contrast, the coding operator $\mathbf{I}_2$ in Theorem 4.1 is exactly optimal in every source despite these changing conditional probabilities.

C.5 The domain of admissible coding operators

Equation (15) is a representation-dependent constraint. It requires unit output norm on the fixed input domain, as in the MCR^2 objective. For a general encoder image it may be more restrictive than for the two-code encoders. The proof of Theorem 4.1 uses only the exact identity in Equation (16) for f_C, f_S and the fact that every admissible $\mathbf{D}f$ remains unit-norm. It does not require a characterisation of $\mathcal{D}(f)$ for every possible encoder.

One can instead apply a linear map and then normalise each output. For the two-code encoder, any nonzero columns again induce an arbitrary pair of unit output vectors after normalisation, giving the same inner optimisation over $(\mathbf{u}, \mathbf{v}) \in \mathbb{S}^1 \times \mathbb{S}^1$. The stated theorem uses Equation (15) to avoid undefined normalisation at zero. It makes no statement about an unconstrained linear operator whose outputs can grow without bound.

D NUMERICAL VALIDATION OF THE THEORETICAL CONSTRUCTIONS

We use finite-sample experiments to examine three predictions of our analysis: coding quality can survive correlation reversal, a small coding gap can coexist with severe prediction failure, and a shared optimal coding operator need not stabilise prediction. The experiments evaluate the stable and environmental encoders defined in Equation (8), and fit the classifier using source samples only. We draw 200,000 independent observations per environment and repeat each setting with 50 random seeds. We sample the four input-cell counts directly from their multinomial law, which is equivalent to sampling individual observations and then counting them. Coding values are computed from empirical second moments and log-determinants, independently of the closed-form reference curves. Unless otherwise stated, error bars denote one standard deviation across seeds. The source classifier minimises empirical 0–1 risk over the two output codes and remains fixed during testing. The coding coefficient is $a = 2/\epsilon^2$.

Correlation reversal preserves coding quality but changes prediction. We fit a classifier in a source environment with mismatch probability 0.01 and hold it fixed while varying the target mismatch probability. Figure 5 shows agreement between the empirical coding values, prediction errors, and theoretical curves. At $a = 2$, the mean coding values at target mismatch probabilities 0.01 and 0.99 are 0.137318 and 0.137250, respectively, compared with the common population value 0.137284. The corresponding mean prediction errors are 0.995% and 98.995%. The stable encoder has zero prediction error throughout. Thus, the coding objective can retain its value while the source label assignment becomes unreliable. The noiseless endpoints have different input support; the positive-noise settings share the same population input support.

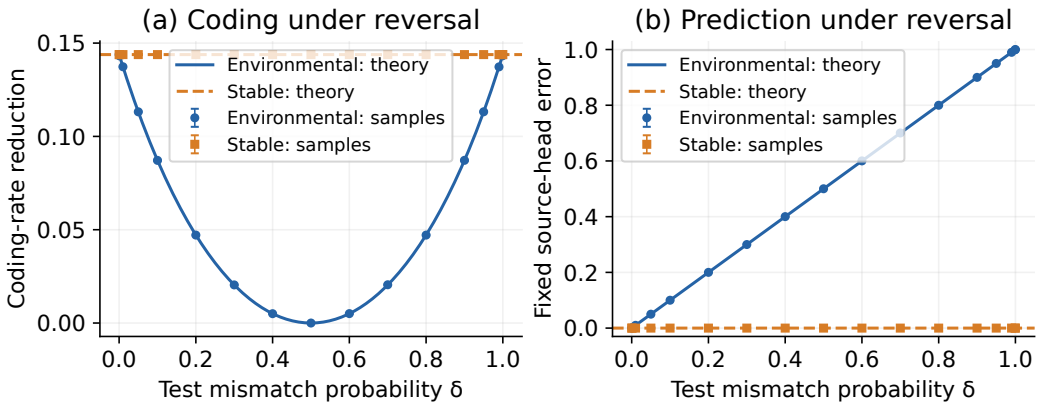

Figure 5: Correlation reversal preserves coding quality while prediction fails. Lines show theory; points show empirical means with one-standard-deviation error bars.

A small coding gap coexists with almost maximal target error. We use three equally weighted source environments with mismatch probabilities $\delta_{\max}/4$, $\delta_{\max}/2$, and $\delta_{\max}$, and a target environment with mismatch probability $1 - \delta_{\max}$. Figure 6 compares the measured coding gap with the theoretical prediction for $a \in \{0.5, 2, 8\}$. As $\delta_{\max}$ decreases, the coding gap decreases while the target error increases. At $a = 2$ and $\delta_{\max} = 0.001$, the mean gap to the population global optimum is 3.868×10^{-4}, compared with the predicted 3.884×10^{-4}, and the mean target error is 99.900%. All four possible inputs occur in every source and target sample in this experiment; the smallest observed cell count is 15. This rules out missing sample support as an explanation of the observed failure in these runs. The benchmark is the population optimum, not an asserted empirical optimum for each randomly imbalanced sample.

A shared optimal coding operator does not stabilise prediction. We next use source mismatch probabilities 0.005, 0.01, and 0.02, with a reversed target at 0.99. For each environment, we evaluate operators with two unit columns whose relative angle ranges from 0° to 180° in 2° increments. This family includes nonorthogonal and rank-one operators, not only rotations. In every seed and

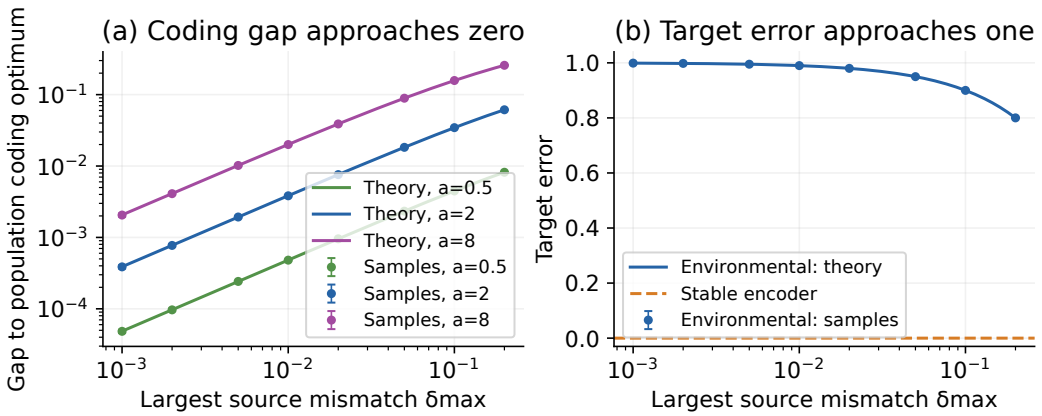

Figure 6: Near-optimal coding and almost maximal target error coexist with identical input support.

environment, the empirical grid maximum occurs at $90°$, consistent with the theorem's shared-optimality result. Nevertheless, the fixed source classifier has approximately 99% target error at the identity operator (Figure 7). Agreement on the best coding operator therefore does not establish a stable label assignment.

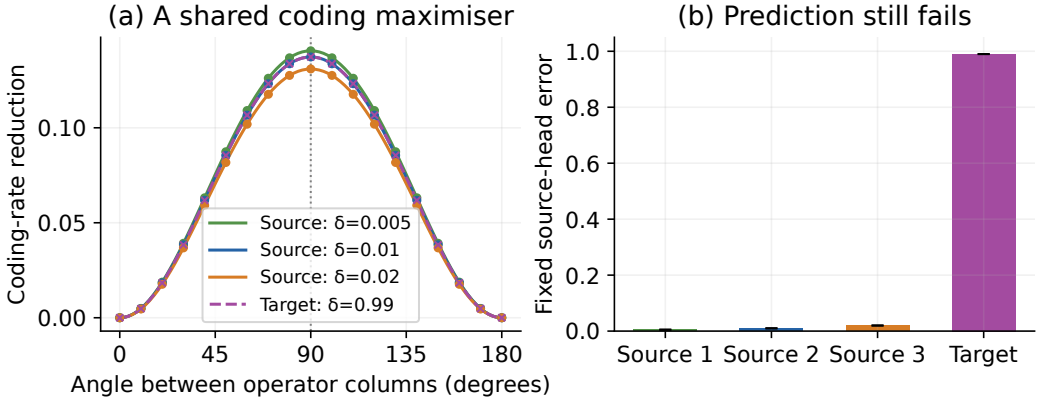

Figure 7: A common coding maximiser coexists with severe target prediction failure.

More samples improve estimation, not target prediction. To separate estimation error from prediction failure, we fix the three source mismatch probabilities at 0.00025, 0.0005, and 0.001, the target mismatch at 0.999, and $a = 2$. We vary the sample size per environment from 10^3 to 10^6, again using 50 independent seeds at each size. Figure 8 shows the root mean squared error (RMSE) of the empirical coding gap and target error relative to their population predictions, computed across seeds. The coding-gap RMSE decreases from 2.82×10^{-4} to 1.10×10^{-5}, and the target-error RMSE decreases from 1.05×10^{-3} to 3.02×10^{-5}. Meanwhile, mean target error remains between 99.874% and 99.905%, close to the predicted 99.9%. The stable encoder has zero error in every run.

Small samples can miss rare inputs even though the population support is shared. We therefore record whether all four inputs are observed in every source and target environment, without rejecting or resampling incomplete draws. This occurs in $0/50$, $21/50$, $50/50$, and $50/50$ runs at the four increasing sample sizes. At 10^6 samples, the least frequent cell still contains at least 103 observations in every environment and run. Thus, in these larger samples, severe target failure persists even when every possible input has been observed. Increasing sample size improves agreement with the theory without repairing the unstable predictive relationship.

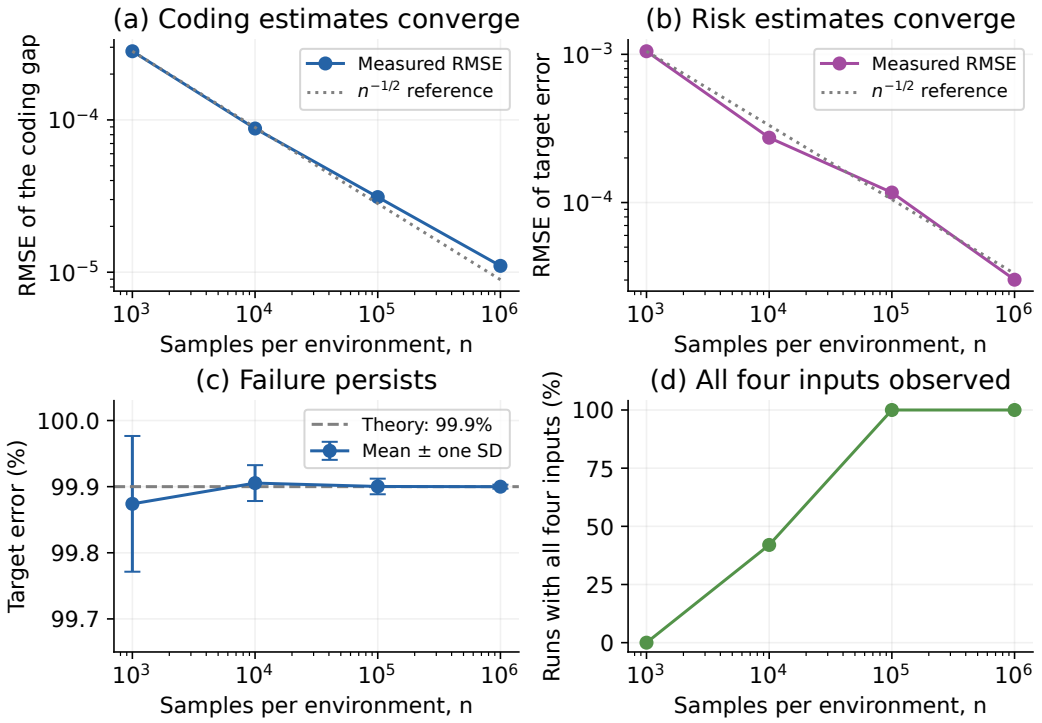

Figure 8: Larger samples improve estimation while target failure persists. RMSE is measured against the population predictions. Dotted lines in (a,b) are $n^{-1/2}$ references, not fitted rates. Panel (c) shows means and one standard deviation; panel (d) reports the fraction of runs observing every input in all four environments.

Exact and near-optimal coding describe different cases. We also check the noiseless boundary separately, with source mismatch zero and target mismatch one. Across 50 runs of 200,000 samples per environment, the stable and environmental encoders have identical empirical source coding values, while the environmental source classifier has zero source error and 100% target error. In this case, the source and target supports differ. Table 2 contrasts this boundary with the positive-noise, identical-support construction. The exact global optimality statement is a population result from the theory; numerical samples need not have an exactly balanced label prior.

Table 2: The two failure regimes. Coding guarantees refer to the population; errors are empirical means over 50 runs of 200,000 samples. The positive-noise row uses $\delta_{\max} = 0.001$ and $a = 2$.

Setting	Input support	Coding guarantee	Target error
Noiseless reversal	Changes	Exactly optimal	100%
Positive noise	Identical	Near-optimal	99.900%

Together, these experiments show agreement between the theoretical predictions and finite-sample measurements. They evaluate prescribed encoders, so they do not establish how frequently an encoder-training algorithm selects an unstable representation. The shared-operator experiment concerns the constraint in Equation (17), rather than other formulations of invariance.

E WATERBIRDS EXPERIMENTAL SETUP

This appendix describes the Waterbirds data pipeline, encoder training, linear-probe evaluation, and saliency computation in the supplied implementation. Hyperparameters listed below are the defaults of `Train-MCR2.py` in the anonymous supplementary code package.

Dataset and environment construction. Waterbirds (Sagawa et al., 2020a) provides bird images with land or water backgrounds. We use the supplied images and metadata rather than generating new composites or assigning new backgrounds. The bird label is $y \in \{0, 1\}$, with 0 denoting landbirds and 1 denoting waterbirds; the background label is $p \in \{0, 1\}$, with 0 denoting land and 1 denoting water. The metadata define four groups (y, p). In the training split, approximately 95% of each bird class has a matching background ($p = y$), and the remaining 5% has a mismatched background ($p \neq y$). All four groups are pooled into a single training dataset. The encoder loss uses bird labels y; background labels p are used to define evaluation groups. We retain the original metadata split assignments: `split=0` for training and `split=2` for testing. The implementation does not use the validation split.

The background-mismatched evaluation uses the two existing test groups $(y, p) = (0, 1)$ and $(1, 0)$, containing $2{,}255$ and 642 images, respectively. This evaluates images whose background association opposes the predominant training association; it does not modify test images or create a new test split. The original test split can be evaluated in full and the two groups reported separately, or filtered to these same $2{,}897$ images before evaluation.

Image preprocessing. During encoder training, we apply a random resized crop to 224×224 pixels, with crop area sampled from $[0.8, 1.0]$ of the original image area, followed by a random horizontal flip with probability 0.5. For feature extraction, testing, and saliency computation, we resize the shorter image side to 256 pixels and take a 224×224 centre crop. RGB values are converted to tensors and normalised using channel means $(0.485, 0.456, 0.406)$ and standard deviations $(0.229, 0.224, 0.225)$.

Encoder and training objective. The encoder is a randomly initialised ResNet-18 (He et al., 2016), with no pretrained weights. Its original classifier is replaced by a linear projection from the 512-dimensional pooled backbone output to a 128-dimensional feature vector. The projection weights use Xavier uniform initialisation and its bias is initialised to zero. Each output feature is normalised to satisfy $\|\mathbf{z}_i\|_2 = 1$. The backbone and projection are trained jointly using the negative coding-rate reduction, with equal weights on the expansion and compression terms. The compression term is computed separately for the two bird classes within each minibatch and weighted by their minibatch proportions. The coding precision is $\epsilon = 1$. In the code, `eps` denotes the denominator parameter corresponding to ϵ^2 in Equation (1); both equal one here. A diagonal term $10^{-6}\mathbf{I}$ is added when computing each log-determinant.

Optimisation settings. The selected script defaults to 50 training epochs, a minibatch size of 128, and AdamW with learning rate 3×10^{-4} and weight decay 10^{-4}. Training data are shuffled each epoch, and the final incomplete minibatch is dropped. The total gradient norm is clipped to 10. Although a cosine learning-rate scheduler is instantiated, it is never stepped in the supplied training loop, so the effective learning rate remains constant. The PyTorch and NumPy random seeds are set to 42, and data loading uses four workers by default. The final encoder checkpoint is saved after training.

Linear-probe evaluation. After each training epoch, the encoder is fixed and placed in evaluation mode. We extract features from all training images using the deterministic preprocessing above and fit a fresh binary logistic-regression classifier to these features and their bird labels. Its weight and bias are initialised to zero. The probe minimises the mean binary cross-entropy plus $\frac{\lambda}{2}\|\mathbf{w}\|_2^2$, where $\lambda = 2 \times 10^{-3}$; the bias is not regularised. Optimisation uses L-BFGS with learning rate 1, at most 200 iterations, and a strong Wolfe line search. Predictions are waterbird when the logit is positive and landbird otherwise. Probe fitting does not update the encoder.

Accuracy is computed separately within each (y, p) test group. For the two background-mismatched groups in Table 1, overall accuracy is the proportion of correct predictions among their combined $2{,}897$ images, rather than the unweighted mean of the two group accuracies. The implementation records test metrics after each epoch but does not implement checkpoint selection based on test accuracy.

Saliency computation. For a deterministically preprocessed training image $\mathbf{x}$, saliency is computed from the encoder outputs rather than the linear-probe logits. At each pixel (u, v), the default map is

$$S(u, v) = \max_{c \in \{1,2,3\}} \sum_{k=1}^{128} \left| \frac{\partial z_k(\mathbf{x})}{\partial x_{c,u,v}} \right| . \tag{75}$$

Thus, absolute input gradients are summed over all feature dimensions and then maximised over RGB channels. Map values are clipped at their 99.95th percentile and rescaled to $[0, 1]$, then overlaid on the image with opacity 0.45. The script generates maps for the first ten training images of each class in dataset order; Figure 3 displays one example from each class.

Software and execution. The implementation uses PyTorch and torchvision for the model and data pipeline, NumPy for numerical utilities, Pillow for image loading, and Matplotlib for saliency visualisation. It automatically uses CUDA when available and otherwise runs on the CPU. The supplementary package specifies compatible dependency ranges rather than the exact software versions of the historical run; it does not include hardware specifications, historical run logs, or pretrained checkpoints.