Maximally Distant Cross Domain Generators
for Estimating Per-Sample Error

Sagie Benaim∗
Tel Aviv University
Tomer Galanti∗
Tel Aviv University
Lior Wolf
Facebook AI Research, Tel Aviv University

Abstract

While in supervised learning, the validation error is an unbiased estimator of the generalization (test) error and complexity-based generalization bounds are abundant, no such bounds exist for learning a mapping in an unsupervised way. As a result, when training GANs and specifically when using GANs for learning to map between domains in a completely unsupervised way, one is forced to select the hyperparameters and the stopping epoch by subjectively examining multiple options. We propose a novel bound for predicting the success of unsupervised cross domain mapping methods, which is motivated by the recently proposed simplicity hypothesis. The bound can be applied both in expectation, for comparing hyperparameters, or per sample, in order to predict the success of a specific cross-domain translation. The utility of the bound is demonstrated in an extensive set of experiments employing multiple recent algorithms.

1. Introduction

The process of selecting hyperparameters and the lack of clear stopping criteria in unsupervised learning are a constant source of frustration. This issue is commonplace for GANs [6] and the derived technologies, in which the training process optimizes multiple losses that balance each other. Practitioners are often uncertain regarding the results obtained when evaluating GAN-based methods, and many avoid using these altogether. One solution is to employ more stable methods such as [2]. However, these methods do not always match the results obtained by GANs. In this work, we offer, for an important family of GAN methodologies, an algorithm for selecting the hyperparameters, as well as a stopping criterion.

Specifically, we focus on predicting the success of algorithms that map between two image domains in an unsupervised manner. Recently, multiple GAN-based methods have demonstrated convincing results, despite the apparent inherent ambiguity, which is described in Sec. 2. We derive what is, as far as we know, the first error bound for unsupervised cross domain mapping. It is inspired by the recently proposed simplicity hypothesis [18]. The success of our method forms an additional source of empirical validation of the theory, beyond the predictions that were presented and validated in [18].

In addition to the novel capability of predicting the success of algorithms that map between two image domains in an unsupervised manner. Recently, multiple GAN-based methods have demonstrated convincing results, despite the apparent inherent ambiguity, which is described in Sec. 2. We derive what is, as far as we know, the first error bound for unsupervised cross domain mapping. It is inspired by the recently proposed simplicity hypothesis [18]. The success of our method forms an additional source of empirical validation of the theory, beyond the predictions that were presented and validated in [18].

In addition to the novel capability of predicting the success, in expectation, of a mapping that was trained using one of the unsupervised mapping methods, we are able to predict the success of mapping every single sample individually. This is remarkable for two reasons: (i) even supervised generalization bounds do not deliver this capability; and (ii) we deal with complex multivariate re-
gression problems (mapping between images) and not with classification problems, in which pseudo probabilities are often assigned.

In Sec. 2, we formulate the problem and present background on the simplicity hypothesis of [18]. Then, in Sec. 3, we derive the prediction bounds. Sec. 4 presents extensive empirical evidence for the success of our algorithms when applied to multiple recent methods.

1.1. Previous work

Generative Adversarial Networks GAN [6] methods train a generator network $G$ that synthesizes samples from a target distribution, given noise vectors, by jointly training a second, adversarial, network $D$. The specific generative architecture underlying the methods we experiment with, is based on the architecture of [12].

Conditional GANs employ a vector of parameters that directs the generator, in addition to (or instead of) the noise vector. These GANs can generate images from a specific class [11] or based on a textual description [15], or invert mid-level network activations [4]. Our bound also applies in these situations. However, this is not the focus of our experiments, which target image mapping, in which the created image is based on an input image [8, 25, 22, 9, 17, 7, 1].

Unsupervised Mapping The validation of our bound focuses on recent cross-domain mapping methods that employ no supervision except for sample images from the two domains. This ability was demonstrated very recently [8, 25, 22, 1] in image to image translation and slightly earlier for translating between natural languages [20].

The DiscoGAN [8] method that we experiment with, similar to other methods [25, 22], learns mappings in both directions, i.e., from domain $A$ to domain $B$ and vice versa. Our experiments also employ the DistanceGAN method [1], which unlike the circularity based methods, is applied only in one direction (from $A$ to $B$). The constraint used by this method is that the distances for a pair of inputs $x_1, x_2 \in A$ before and after the mapping, by the learned mapping $G$, are highly correlated, i.e.,

$$||x_1 - x_2|| \sim ||G(x_1) - G(x_2)||.$$  \tag{1}

Weakly Supervised Mapping Our bound can also be applied to GAN based methods that match between the source domain and the target domain by also incorporating a fixed pre-trained feature map $f$ and requiring $f$-constancy, i.e., that the activations of $f$ are the same for the input samples and for mapped samples [17, 19]. During training, the various components of the loss (GAN, $f$-constancy, and a few others) do not provide a clear signal when to stop training or which hyperparameters to use.

Generalization Bounds for Unsupervised Learning Only a few generalization bounds for unsupervised learning were suggested in the literature. In [16], PAC-Bayesian generalization bounds are presented for density estimation. In [14], an algorithm is shown for estimating a bounded density using a finite combination of densities from a given class. It is proven that the estimation error of this algorithm is bounded by $O(1/\sqrt{n})$. Our work studies the error of a mapping and not the KL-divergence with respect to a target distribution, and our bound is data-dependent and not based on the complexity of the hypothesis class.

2. Problem Setup

The learning algorithm is provided with two unlabeled datasets: one includes i.i.d samples from a first distribution and the second includes i.i.d samples from a second distribution.

$$x_i \in \mathcal{X}_A \text{ for } i = 1 \ldots m \text{ where } x_i \overset{i.i.d}{\sim} D_A$$

and $\mathcal{X}_A$ denotes the space of domain $A = (\mathcal{X}_A, D_A)$

$$x_j \in \mathcal{X}_B \text{ for } j = 1 \ldots n \text{ where } x_j \overset{i.i.d}{\sim} D_B$$

and $\mathcal{X}_B$ denotes the space of domain $B = (\mathcal{X}_B, D_B)$. \tag{1}

We denote the two training datasets by $S_A$ and $S_B$ (resp.), and $y_{AB}$ denotes the target function, which is one of the functions that map the first domain to the second, such that $y_{AB} \circ D_A = D_B$. The goal of the learner is to fit a function $G \in \mathcal{H}$, for some hypothesis class $\mathcal{H}$ that is closest to $y_{AB}$, i.e.,

$$\inf_{G \in \mathcal{H}} R_{D_A}[G, y_{AB}]$$ \tag{2}
where $R_D[f_1, f_2] = \mathbb{E}_{x \sim D} [\ell(f_1(x), f_2(x))]$, for a loss function $\ell : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ and distribution $D$.

It is not clear that such fitting is possible without further information. Assume, for example, that there is a natural order on the samples in $X_B$. A mapping that maps an input sample $x \in X_A$ to the sample that is next in order to $y_{AB}(x)$ could be just as feasible. More generally, one can permute the samples in $X_A$ by some function $\Pi$ that replaces each sample with another sample that has a similar likelihood and learn $G$ that satisfies $G = \Pi \circ y_{AB}$. This difficulty is referred to in [18] as “the alignment problem”.

In multiple recent contributions [20, 8, 12, 22] circularity is employed. Circularity requires the recovery of both $y_{AB}$ and $y_{BA} = y_{AB}^{-1}$ simultaneously. Namely, functions $G$ and $G'$ are learned jointly by minimizing the following objective:

$$
\text{disc}(G \circ D_A, D_B) + \text{disc}(G' \circ D_B, D_A) + R_{D_A}[G' \circ G, \text{Id}_A] + R_{D_B}[G \circ G', \text{Id}_B] 
$$

(3)

where

$$
\text{disc}(D_1, D_2) = \sup_{c_1, c_2 \in C} \left| R_{D_1}[c_1, c_2] - R_{D_2}[c_1, c_2] \right|
$$

(4)

denotes the discrepancy between distributions $D_1$ and $D_2$, and $C$ is a set of discriminators. This discrepancy is implemented by a GAN, as in [5].

As shown in [18], the circularity constraint does not eliminate the uncertainty in its entirety, and is secondary to the GANs themselves. In Distance-GAN [1], the circularity was replaced by a multi-dimensional scaling type of constraint, which enforces a high correlation between the distances in the two domains. However, since these constraints hold only approximately, the ambiguity is not completely eliminated.

2.1. The Simplicity Hypothesis

In order to understand how the recent unsupervised image mapping methods work despite the inherent ambiguity, [18] recently hypothesized that the target (“semantic”) mapping $y_{AB}$ is typically the distribution preserving mapping with the lowest complexity. It was shown that such mappings are expected to be unique.

2.2. Occam’s Razor

As a motivating example to the key role of minimal mappings, consider, following [18], the domain $A$ of uniformly distributed points $(x_1, x_2) \in \mathbb{R}^2$, where $0 \leq x_1 < 1$ and $x_2 = 1$. Let $B$ be a similar domain, except $x_2 = 2$. We note that there are infinitely many mappings from domain $A$ to $B$ that, given inputs in $A$, result in the uniform distribution of $B$ and satisfy the circularity constraint (Eq. 3).

However, it is easy to see that when restricting the hypothesis class to neural networks with one layer of size 2, and Leaky ReLU activations $\sigma$ [10], there are only two options left. In this case, $h(x) = \sigma(Wx + b)$, for $W \in \mathbb{R}^{2 \times 2}, b \in \mathbb{R}^2$. The only admissible solutions are of the form $W_a = \begin{pmatrix} 1 & -b_1 \\ 0 & 2 - b_2 \end{pmatrix}$ or $W_b = \begin{pmatrix} -1 & 1 - b_1 \\ 0 & 2 - b_2 \end{pmatrix}$ and $b = (b_1, b_2)^T$, which are identical, for every $b$, to $y_{AB}^0((x_1, 1)^T) = (x_1, 2)^T$ or to the alternative $y_{AB}^0((x_1, 1)^T) = (1 - x_1, 2)^T$.

Therefore, by restricting the hypothesis space to be minimal, we eliminate all alternative solutions, except two. These two are exactly the two mappings that would commonly be considered “more semantic” than any other mapping, see Fig. 1.

![Figure 1. An illustrative example where the two domains are line segments in $\mathbb{R}^2$. There are infinitely many mappings that preserve the uniform distribution on the two segments. However, only two stand out as “semantic”. These are exactly the two mappings that can be captured by a neural network with a single layer of size 2 and Leaky ReLU activations.](image-url)
known as Occam’s razor. Our formulation of this principle is more general than the framework used in [18], which was limited to a Kolmogorov-like complexity of multi-layered neural networks.

Given two domains $A = (X_A, D_A)$ and $B = (X_B, D_B)$, a mapping $f : X_A \to X_B$ satisfies the Occam’s razor property between domains $A$ and $B$, if it has minimal complexity among the functions $h : X_A \to X_B$ that satisfy $h \circ D_A \approx D_B$. Minimal complexity is defined by nesting of hypothesis classes, which forms a partial order, and not as a continuous score. For example, if $H_i$ is the set of neural networks of a specific architecture and $H_j$ is the set of neural networks of the architecture obtained after deleting one of the hidden neurons, then, $H_i \subset H_j$. Intuitively, minimal complexity would mean that there is no sub-class that can implement a mapping $f' : X_A \to X_B$ such that $f' \circ D_A \approx D_B$. For this purpose, we define,

$$\mathcal{P}(H; \epsilon) = \left\{ \hat{G} \in H \mid \text{disc}(G \circ D_A, D_B) \leq \epsilon \right\}$$

**Definition 1.** Let $A = (X_A, D_A)$ and $B = (X_B, D_B)$ be two domains and $U = \{H_i\}_{i \in I}$ be a family of hypothesis classes. A mapping $y_{AB} : X_A \to X_B$ satisfies an $(\epsilon_1, \epsilon_2)$-Occam’s razor property if for all $H \in U$, we have: $\mathcal{P}(H; \epsilon_1) \neq \emptyset$ implies $\inf_{G \in \mathcal{P}(H; \epsilon_1)} R_{DA}[G, y_{AB}] \leq \epsilon_2$.

Uniqueness is not assumed: if $y_{AB}$ has the $(\epsilon_1, \epsilon_2)$-Occam’s razor property, then it will be $\epsilon_2$ close to a function in every minimal hypothesis class $H \in U$ that has a non-empty $\mathcal{P}(H; \epsilon_1)$. As the hypothesis class grows, so does $\mathcal{P}(H; \epsilon_1)$, i.e., $H_i \subset H_j$ implies that $\mathcal{P}(H_j; \epsilon_1) \subset \mathcal{P}(H_i; \epsilon_1)$. The growing $\mathcal{P}(H; \epsilon_1)$ would always contain $y_{AB}$.

### 3. Estimating the Ground Truth Error

Our method bounds the generalization risk $R_{DA}[G_1, y_{AB}]$, i.e., the expected loss of the learned mapping $G_1$. The quantity used to bound it is based on two terms and holds for every $\epsilon_1$. The first term is the maximal risk possible with a member of the class $\mathcal{P}(H; \epsilon_1)$. The second term in the bound is the approximation error of $y_{AB}$ within $H$.

The bound is a consequence of using a loss $\ell$ that satisfies the triangle inequality. Losses of this type include the $L_1$ loss, which is often used in cross domain mapping. The $L_2$ loss satisfies the triangle inequality up to a factor of three, which would incur the addition of a factor into the bound.

**Lemma 1.** Let $A = (X_A, D_A)$ and $B = (X_B, D_B)$ be two domains. In addition, let $\ell$ be a loss function satisfying the triangle inequality, $\mathcal{H}$ be a hypothesis class, and let $y_{AB}$ and $G_1$ be any two functions. Then,

$$R_{DA}[G_1, y_{AB}] \leq \sup_{G_2 \in \mathcal{P}(H; \epsilon_1)} R_{DA}[G_1, G_2] + \inf_{G \in \mathcal{P}(H; \epsilon_1)} R_{DA}[G, y_{AB}]$$

and,

$$\sup_{G_2 \in \mathcal{P}(H; \epsilon_1)} R_{DA}[G_1, G_2] \leq R_{DA}[G_1, y_{AB}] + \sup_{G \in \mathcal{P}(H; \epsilon_1)} R_{DA}[G, y_{AB}]$$

**Proof.** First, we prove Eq. 6. Let $G^* = \arg \inf_{G \in \mathcal{P}(H; \epsilon_1)} R_{DA}[G, y_{AB}]$. By the triangle inequality, we have:

$$R_{DA}[G_1, y_{AB}] \leq R_{DA}[G_1, G^*] + R_{DA}[G^*, y_{AB}] \leq \sup_{G_2 \in \mathcal{P}(H; \epsilon_1)} R_{DA}[G_1, G_2] + \inf_{G \in \mathcal{P}(H; \epsilon_1)} R_{DA}[G, y_{AB}]$$

In order to prove Eq. 7, we use the triangle inequality as follows:

$$\sup_{G_2 \in \mathcal{P}(H; \epsilon_1)} R_{DA}[G_1, G_2] \leq \sup_{G_2 \in \mathcal{P}(H; \epsilon_1)} (R_{DA}[G_1, y_{AB}] + R_{DA}[G_2, y_{AB}]) = R_{DA}[G_1, y_{AB}] + \sup_{G_2 \in \mathcal{P}(H; \epsilon_1)} R_{DA}[G_2, y_{AB}]$$

If $y_{AB}$ satisfies Occam’s razor, then the approximation error is lower than $\epsilon_2$ and the following bound is obtained:

$$R_{DA}[G_1, y_{AB}] \leq \sup_{G_2 \in \mathcal{P}(H; \epsilon_1)} R_{DA}[G_1, G_2] + \epsilon_2$$

Eq. 8 provides us with an accessible bound for the generalization risk. The RHS can be directly approximated by training a neural network.
Algorithm 1 Deciding when to stop training $G_1$

Require: $S_A$ and $S_B$: unlabeled training sets;
$H$: a hypothesis class $H$; $\epsilon_1$: a threshold; $\lambda$: a trade-off parameter; $T_2$: a fixed number of epochs for $G_2$; $T$: a maximal number of epochs.
1: Initialize $G_1^0$.
2: for $i = 1, \ldots, T$ do
3: Train $G_1^{i-1}$ for one epoch to minimize $\text{disc}(G_1 \circ D_A, D_B)$, obtaining $G_1^i$.
4: Train $G_2$ for $T_2$ epochs to minimize $\text{disc}(G_2 \circ D_A, D_B) - \lambda R_{D_A}[G_1^i, G_2]$.
\> $T_2$ provides a fixed comparison point.
5: end for
6: return $G_1^t$ such that: $t = \arg\min_{i \in [T]} R_{D_A}[G_1^i, G_2^i]$.

Algorithm 2 Model Selection

Require: $S_A$ and $S_B$: unlabeled training sets; $U = \{H_\lambda\}_{\lambda \in I}$: a family of hypothesis classes; $\epsilon$: a threshold; and $\lambda$: a trade-off parameter.
1: Initialize $J = \emptyset$.
2: for $i \in J$ do
3: Train $G_1^i \in H_\lambda$ to minimize $\text{disc}(G_1^i \circ D_A, D_B)$.
4: if $\text{disc}(G_1^i \circ D_A, D_B) \leq \epsilon$ then
5: Add $i$ to $J$.
6: Train $G_2$ to minimize $\text{disc}(G_2 \circ D_A, D_B) - \lambda R_{D_A}[G_1^i, G_2]$.
7: end if
8: end for
9: return $G_1^i$ such that: $i = \arg\min_{i \in J} R_{D_A}[G_1^i, G_2^i]$.

Algorithm 3 Bounding the loss of $G_1$ on sample $x$

Require: $S_A$ and $S_B$: unlabeled training sets; $G_1$: a mapping; $H$: a hypothesis class; $\lambda$: a trade-off parameter; $x$: a specific sample $x$.
1: Train $G_2$ to minimize $\text{disc}(G_2 \circ D_A, D_B) - \lambda(f(G_1(x), G_2(x)))$.
2: return $\ell(G_1(x), G_2(x))$.

Lemma 2. Let $A = (X_A, D_A)$ and $B = (X_B, D_B)$ be two domains. In addition, let $\ell$ be a loss function satisfying the triangle inequality and $H$ be a hypothesis class. Let $y_{AB}$ be a target function and $G_1$ be any function. Then,

$$\ell(G_1(x), y_{AB}(x)) \leq \sup_{G_2 \in P(H_{\lambda})} \ell(G_1(x), G_2(x)) + \inf_{G \in P(H_{\lambda})} \ell(G(x), y_{AB}(x))$$

In practice, we run a modified Alg. 3, in which $x$ is weighted to half the weight of all samples, during the training of $G_2$. This emphasizes the role of $x$ and allows us to train $G_2$ for less epochs, which is important since a different $G_2$ network is trained for measuring the error of each sample.

4. Experiments

We test the three algorithms on two unsupervised alignment methods: DiscoGAN [8] and DistanceGAN [1]. In DiscoGAN, we train $G_1$, the classifier to be evaluated, using two GANs and two circularity constraints; in DistanceGAN, one GAN and one distance correlation loss are used. The pub-
Alg. Method Dataset Bound GANA GANB CycleA/L∗ CycleB

Alg. 1 Disco- Shoes2Edges 1.00 (<1E-16) -0.15 (3E-03) -0.28 (1E-08) -0.43 (3E-01) -0.25 (9E-05) 1.00 (<1E-16) 0.25 (1E-01) 0.08 (6E-01) 0.08 (6E-01) 0.05 (3E-01)

Alg. 2 Disco- Bags2Edges 0.99 (2E-06) 0.64 (2E-01) 0.54 (3E-01) -0.26 (7E-01) -0.20 (7E-01)
Cityscapes 0.99 (1E-03) 0.69 (9E-02) 0.85 (2E-02) -0.53 (2E-01) -0.42 (4E-01)
Facades 0.94 (1E-03) -0.33 (4E-01) 0.88 (4E-02) 0.66 (8E-02) -0.45 (3E-01)
Maps 1.00 (1E-03) 0.62 (1E-01) 0.54 (2E-01) 0.62 (2E-01) 0.60 (2E-01) 0.67 (2E-01)

Alg. 3 Disco- Bags2Edges 0.92 (1E-16) 0.12 (5E-01) 0.02 (9E-01) 0.29 (6E-02) 0.15 (4E-01)
Cityscapes 0.78 (4E-04) 0.24 (4E-01) -0.16 (6E-01) -0.04 (9E-01) 0.03 (9E-01)
Facades 0.80 (4E-10) 0.13 (4E-01) 0.16 (3E-01) 0.20 (2E-01) 0.09 (5E-01)
Maps 0.66 (1E-03) 0.08 (7E-01) 0.12 (6E-01) 0.17 (5E-01) -0.25 (3E-01)

Dist- Shoes2Edges 0.96 (1E-04) - - 0.33 (5E-01) 0.20 (9E-05) -0.14 (5E-03)
Dist- Bags2Edges 0.98 (1E-05) - -0.11 (8E-01) 0.23 (6E-01) -
Cityscapes 0.92 (1E-03) - -0.66 (8E-02) -0.49 (2E-01) -
Facades 0.84 (2E-02) - 0.75 (5E-02) 0.37 (4E-01) -
Maps 0.95 (1E-03) - -0.43 (3E-01) -0.15 (7E-01) -

Table 1. Pearson correlations and the corresponding p-values (in parentheses) of the ground truth error with: (i) the bound, (ii) the GAN losses, and (iii) the circularity losses or (iv) the distance correlation loss. The cycle loss $A \rightarrow B \rightarrow A$ is shown for DiscoGAN and the distance correlation loss is shown for DistanceGAN.

Five datasets were used in the experiments: (i) Aerial photographs to maps, trained on data scraped from Google Maps [7]. (ii) The mapping between photographs from the cityscapes dataset and their per-pixel semantic labels [3]. (iii) Architectural photographs to their labels from the CMP Facades dataset [13]. (iv) Handbag images [24] to their binary edge images as obtained from the HED edge detector [21]. (v) A similar dataset for the shoe images from [23].

The results of all experiments are summarized in Tab. 1, which presents the correlation and p-value between the ground truth error rate as a function of the independent variable and the bound. The independent variable is either the training epoch, the architecture, or the sample, depending on the algorithm tested. Similar correlations are shown with the GAN losses and the reconstruction losses (DiscoGAN) or the distance correlation loss (DistanceGAN), in order to convince that these are much less correlated with the ground truth error. In the plots below, we omit the other scores in order to reduce clutter.

**Stopping Criterion (Alg. 1)** For testing the stopping criterion suggested in our first algorithm, we
Figure 2. Results of Alg. 1, 2. The ground truth errors and the bound are shown to be highly correlated. x-axis is iteration or number of layers, depending on the experiment. y-axis is the expected risk. Note that in Alg. 1 it takes a few epochs for $G_1$ to have a small enough discrepancy, until which the bound is ineffective.

compared, at each time point, two scores that are averaged over all training samples: $||G_1(x) - G_2(x)||_1$, which is our bound, and the ground truth error $||G_1(x) - y_{AB}(x)||_1$, where $y_{AB}(x)$ is the ground truth image that matches $x$ in domain $B$.

Note that similar to the experiments with ground truth in the literature [8, 25, 1], the ground truth error is measured in the label space and not in the photograph domain. The mapping in the other direction $y_{BA}$ is not one to one.

The results are depicted in the main results table (Tab. 1) as well as in Fig. 2 for both DiscoGAN (first column) and DistanceGAN (second column). As can be seen, there is an excellent match between the average ground truth error of the learned mapping $G_1$ and the predicted error. No such level of correlation is present when considering the GAN losses or the reconstruction losses (for DiscoGAN), or the distance correlation loss of DistanceGAN. In this experiment, which has a large number of sample points, the cycle from $B$ to $A$ and back to $B$ is significantly correlated with the ground truth error.
Figure 3. Results of Alg. 3. The first row is of DiscoGAN and the second is of DistanceGAN. The ground truth errors (x-axis) vs. the bound (y-axis) are shown per point.

Figure 4. Results of Alg. 3. The ground truth errors vs. the bound per point are shown on the right. The source, mapping and ground truth of the marked points are shown on the left.

with very low p-values in four out of five datasets. However, its correlation is significantly lower than that of our bound.

Selecting Architecture using Alg. 2 We next consider the effect of changing the number of layers in the network $G$ on the risk. A large correlation with our bound is also observed here, see Tab. 1 and Fig. 2, columns three and four. With a much smaller number of sample points, the p-values are generally higher than in the previous experiment.

Predicting per-Sample Loss with Alg. 3 Finally, we consider the per sample loss. The results are reported numerically in Tab. 1 and plotted in Fig. 3, 4. As can be seen, there is a high degree of correlation between the measured bound and the ground truth error. Therefore, our method is able to reliably predict the per-sample success of a multivariate mapping learned in a fully unsupervised manner.

Remarkably, this correlation also seems to hold when considering the time axis, i.e., we can combine Alg. 1 and Alg. 3 and select the stopping epoch
Figure 5. Per-epoch per-sample results for three experiments, four points each. x-axis is iteration, y-axis is the per-sample error. Red line indicates the ground truth error of an individual sample, i.e. \( |G_1(x) - y(x)| \). Black line indicates our bound for an individual sample, i.e. \( |G_1(x) - G_2(x)| \). Note that it takes a few epochs for \( G_1 \) to have a small enough discrepancy, until which the bound is ineffective.

that is best for a specific sample. Fig. 5 depicts, for three experiments, the bound and the per-sample loss of \( G_1 \) over time. In each graph, we plotted the values of the bound and the loss over time during training of \( G_1 \). In each column we have the results for four samples with a specific dataset and method. As can be seen, in the datasets tested, the bound holds over time. However, the points of a specific dataset seem to follow relatively similar patterns of improvement in time.

5. Conclusions

We extend the envelop of what is known to be possible in unsupervised learning by showing
that we can reliably predict the error of a cross-domain mapping that was trained without matching samples. This is true both in expectation, with application to hyperparameter selection, and per sample, thus supporting dynamic confidence-based run time behavior, and (future work) unsupervised boosting during training.

The method is based on measuring the maximal mutual risk within the set $\mathcal{P}$ of low discrepancy mappings. By applying what we define as the Ockam’s razor property, we bound the approximation error of the ground truth mapping for this set. The clear empirical success we observe supports the recent hypothesis that simplicity plays a role in unsupervised learning. Otherwise, the approximation error would change as the set $\mathcal{P}$ grows (e.g., by adding layers to the network), which would mask the correlation between the maximal mutual risk within $\mathcal{P}$ and the ground truth error.

**Acknowledgements**

This project has received funding from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant ERC CoG 725974).

**References**

[1] S. Benaim and L. Wolf. One-sided unsupervised domain mapping. In *NIPS*, 2017.

[2] P. Bojanowski, A. Joulin, D. Lopez-Paz, and A. Szlam. Optimizing the latent space of generative networks. *arXiv preprint arXiv:1707.05776*, 2017.

[3] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016.

[4] A. Dosovitskiy and T. Brox. Generating images with perceptual similarity metrics based on deep networks. *arXiv preprint arXiv:1602.02644*, 2016.

[5] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky. Domain-adversarial training of neural networks. *J. Mach. Learn. Res.*, 17(1):2096–2030, 2016.

[6] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *NIPS*, pages 2672–2680, 2014.

[7] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017.

[8] T. Kim, M. Cha, H. Kim, J. Lee, and J. Kim. Learning to discover cross-domain relations with generative adversarial networks. *arXiv preprint arXiv:1703.05192*, 2017.

[9] M.-Y. Liu and O. Tuzel. Coupled generative adversarial networks. In *NIPS*, pages 469–477, 2016.

[10] A. L. Maas, A. Y. Hannun, and A. Y. Ng. Rectifier nonlinearities improve neural network acoustic models. In *in ICML Workshop on Deep Learning for Audio, Speech and Language Processing*, 2013.

[11] M. Mirza and S. Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[12] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[13] R. Š. Radim Tyleček. Spatial pattern templates for recognition of objects with regular structure. In *Proc. GCPR*, Saarbrucken, Germany, 2013.

[14] A. Rakhlín, D. Panchenko, and S. Mukherjee. Essaim: Probability and statistics risk bounds for mixture density estimation. 2005.

[15] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee. Generative adversarial text to image synthesis. In *ICML*, 2016.

[16] Y. Seldin and N. Tishby. Pac-bayesian generalization bound for density estimation with application to co-clustering. In *JMLR Workshop and Conference Proceedings Volume 5: AISTATS 2009*, pages 472–479, Cambridge, MA, USA, Apr. 2009. Max-Planck-Gesellschaft, MIT Press.

[17] Y. Taigman, A. Polyak, and L. Wolf. Unsupervised cross-domain image generation. In *International Conference on Learning Representations (ICLR)*, 2017.

[18] L. W. Tomer Galanti and S. Benaim. The role of minimal complexity functions in unsupervised learning of semantic mappings. *arXiv preprint arXiv:1709.00074*, 2017.

[19] L. Wolf, Y. Taigman, and A. Polyak. Unsupervised creation of parameterized avatars. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.
[20] Y. Xia, D. He, T. Qin, L. Wang, N. Yu, T.-Y. Liu, and W.-Y. Ma. Dual learning for machine translation. arXiv preprint arXiv:1611.00179, 2016.

[21] S. Xie and Z. Tu. Holistically-nested edge detection. In The IEEE International Conference on Computer Vision (ICCV), December 2015.

[22] Z. Yi, H. Zhang, P. Tan, and M. Gong. DualGAN: Unsupervised dual learning for image-to-image translation. arXiv preprint arXiv:1704.02510, 2017.

[23] A. Yu and K. Grauman. Fine-grained visual comparisons with local learning. In Computer Vision and Pattern Recognition (CVPR), Jun 2014.

[24] J.-Y. Zhu, P. Krähenbühl, E. Shechtman, and A. A. Efros. Generative visual manipulation on the natural image manifold. In ECCV, 2016.

[25] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. arXiv preprint arXiv:1703.10593, 2017.