Disclaimer: *This work is very much in progress.*

Categories, Functors and Adjunctions for Learned Probability Distributions

1 INTRODUCTION

In this work we describe a category theoretic construction for reasoning about the relationship between probability distributions over R^n and models trained to mimic those distributions. Our construction centers on the relationships between categories of probability distributions and categories of parameter vectors.

While a model can represent a wide range of probability distributions, even a deep neural network (with a fixed set of hyperparameters) cannot model every continuous distribution over R^n. We represent this limitation with a functor that maps probability distributions to generative models that approximate those distributions. We demonstrate that this approximation is adjoint to a functor that maps models' parameters to the distributions they define and use this adjunction to define a Model Fitting Monad. We also formalize the relationship between models trained on samples from a distribution and models configured to optimally approximate that distribution. We conclude with a description of how we can represent the process of mimicking a large model with a smaller one.

2 DISTRIBUTION AND PARAMETER CATEGORIES

Distribution maps over R^n are tuples (D_1, D_2, F) such that D_1 and D_2 are probability distributions over the unit ball in R^n and F is the non-empty set of all functions that map R^n to R^n such that $d \sim D_1$ implies $f(d) \sim D_2$. We note that distribution maps are unique for any D_1, D_2. For the distribution map (D_1, D_2, F), we will refer to D_1 as its **distributional domain** and D_2 as its **distributional codomain**. We will also refer to the elements of F as the **function elements** of (D_1, D_2, F).

- **Example**: If U is the bivariate uniform distribution and N is the standard bivariate normal distribution, then the tuple (U, N, F) is a distribution map over R^2 if the set F contains all functions that accept pairs of uniformly distributed samples and return pairs of standard normally distributed samples, such as the Box-Muller transform function.

For any two distribution maps $M_1 = (D_1, D_2, F_{12}), M_2 = (D_2, D_3, F_{23})$ over R^n such that the distributional codomain of M_1 is equivalent to the distributional domain of M_2, we define the **composition** $M_2 \circ M_1$ to be the distribution map (D_1, D_3, F_{13}) such that F_{13} is the non-empty set of all functions such that $d \sim D_1$ implies $f(d) \sim D_3$.
We can now define the category $Dist_n$ of probability distributions over the unit ball in R^n.

- *Objects*: The objects in $Dist_n$ are probability distributions D over the unit ball in R^n such that there exists some distribution map (U, D, F) with distributional domain equal to the uniform distribution U and distributional codomain equal to D.
- *Arrows*: The arrows in $Dist_n$ are distribution maps such that the source and target objects of an arrow are its distributional domain and codomain.

Lemma 2.1. *$Dist_n$ forms a thin category.*

Proof: First, distribution maps are closed under composition by definition. For the proof of identity, consider the arrow $(D_2, D_2, F_{22}) \circ (D_1, D_2, F_{12}) = (D_1, D_2, G_{12})$. By the definition of distribution maps, $(D_1, D_2, G_{12}) = (D_1, D_2, F_{12})$. By the same argument we see that $(D_2, D_3, F_{23}) \circ (D_2, D_2, F_{22}) = (D_2, D_3, F_{23})$ so the arrow (D_2, D_2, F_{22}) acts as the identity on D_2. Finally, by the definition of a distribution map there must be at most one arrow between any two objects in $Dist_n$, so $Dist_n$ is thin. ∎

We quickly note that $Dist_n$ is not a connected category. Consider the distribution D_{null} that has weight 1 at the zero vector and weight 0 everywhere else. Since every sample drawn from D_{null} is the zero vector, there is no function f that accepts samples drawn from D_{null} and returns samples drawn from U, so the distribution map with distributional domain equal to D_{null} and distributional codomain equal to U does not exist.

Next, we define the category $Dist_{n_c}$ of continuous probability distributions over the unit ball in R^n.

- *Objects*: The objects in $Dist_{n_c}$ are continuous probability distributions D over the unit ball in R^n.
- *Arrows*: The arrows in $Dist_{n_c}$ are distribution maps such that the source and target objects of an arrow are its distributional domain and codomain.

Lemma 2.2. *$Dist_{n_c}$ is a thin and connected category.*

Proof: By the same arguments as Lemma 2.1, $Dist_{n_c}$ forms a thin category. In order to prove that $Dist_{n_c}$ is connected, we will first show that for any continuous probability distribution D, there exists a function f_D such that $d \sim D$ implies $f_D(d) \sim U$. We will then show that there exists an function g_D such that $u \sim U$ implies $g_D(u) \sim D$. This will mean that for any two continuous distributions D_1, D_2, the distribution map (D_1, D_2, F) must exist because $v \in D_1$ implies $g_{D_2} \circ f_{D_1}(v) \in D_2$. This implies $Dist_{n_c}$ is connected.

We first prove the existence of f_D. For any continuous probability distribution D over R^n, we can define the following procedure that accepts

$d \sim D$ and produces $f_D(d) \sim U$:
Input: The n element vector $d \sim D$
Output: The n element vector $u \sim U$
Repeat for $i = 1...n$:

(1) Form F_i, the marginal distribution of D_i conditioned on $d_1, d_2, ..., d_{i-1}$.
(2) Set $u_i = F_i(d_i)$. By the Probability Integral Transform, $u_i \sim U$.

We next prove the existence of g_D. For any continuous probability distribution D over R^n, we can define the following procedure that accepts $u \sim U$ and produces $g_D(u) \sim D$:
Input: The n element vector $u \sim U$
Output: The n element vector $d \sim D$
Repeat for $i = 1...n$:

(1) Form F_i, the marginal distribution of D_i conditioned on the sampled values of $d_1, d_2, ..., d_{i-1}$.
(2) Compute the left-continuous inverse $F_i^*(u) = \inf\{x : F_i(x) = u, 0 < u < 1\}$ and set $d_i = F_i^*(u_i)$. By the Inverse Probability Integral Transform, $d_i \sim D_i$.

Therefore $Dist_{n_c}$ is connected ■

Lemma 2.3. *$Dist_{n_c}$ is a subcategory of $Dist_n$*

Proof: By Lemma 2.2, we see that there is an arrow from the uniform distribution to each object in $Dist_{n_c}$. Therefore the sets of objects and arrows in the category $Dist_{n_c}$ are subsets of the sets of objects and arrows in $Dist_n$. Since $Dist_{n_c}$ forms a category, we see that this implies that $Dist_{n_c}$ is a subcategory of $Dist_n$. ■

We now define a **continuous parametric model** $P_{k \to n}$ to be a function that accepts some vector $(\theta_1, \theta_2, ..., \theta_k) \in R^k$ and returns a continuous probability distribution over the unit ball in R^n such that the the uniform distribution U is in the codomain of $P_{k \to n}$. For some $P_{k \to n}$, we can define the category $Dist_{P_{k \to n}}$:

- *Objects*: The objects in $Dist_{P_{k \to n}}$ are continuous probability distributions D such that there exists some $v \in R^k, P_{k \to n}(v) = D$.
- *Arrows*: The arrows in $Dist_{P_{k \to n}}$ are distribution maps such that the source and target objects of an arrow are its distributional domain and codomain.

Since the objects in $Dist_{P_{k \to n}}$ must be continuous distributions, $Dist_{P_{k \to n}}$ is a subcategory of $Dist_{n_c}$ and is therefore thin and connected.

Next, we define the category $Vect_k$ of vectors and vector addition:

- *Objects*: The objects in $Vect_k$ are vectors in R^k.
- *Arrows*: The arrows between objects v_1 and v_2 are vectors u_{12} such that $v_1 + u_{12} = v_2$.

We define the composition of arrows in $Vect_k$ as vector addition.

Lemma 2.4. *$Vect_k$ is a thin and connected category.*

Proof: First, the zero vector is the identity arrow for every object. Next, if $v_1 + u_{12} = v_2$ and $v_2 + u_{23} = v_3$, then $v_1 + u_{12} + u_{23} = v_2$, so the arrows are closed under composition. Furthermore, since there is always a unique solution to the equation $v_1 + x = v_2, v_1, v_2, x \in R^k$, there must be exactly one arrow between any two objects in $Vect_k$. ■

To summarize, we have defined the following categories:

- $Dist_n$, with probability distributions as objects and distribution maps as arrows.
- $Dist_{n_c}$, with continuous probability distributions as objects and distribution maps as arrows.
- $Dist_{P_{k \to n}}$, with continuous probability distributions defined by the continuous parametric model $P_{k \to n}$ as objects and distribution maps as arrows.
- $Vect_k$, with k-element vectors as objects and arrows such that the composition of arrows is vector addition.

3 OPTIMAL FIT

In this section we introduce functors for relating distributions to the parameters of algorithms that model them. In the case where the model that we choose does not have sufficient capacity to represent the distribution, we define a notion of optimal probability distribution approximation. We build on this to construct an "Optimal Model Fitting" Monad that represents the process of wrapping a probability distribution in a model that approximates it.

3.1 Optimal Fit and Generative Functors

We define a **probability distribution divergence function** L to be a function that accepts two probability distributions over the unit ball in R^n and returns a non-negative value indicating the "difference" between these distributions such that $L(D_1, D_2) = 0$ iff $D_1 = D_2$.
Next, for some continuous parametric model $P_{k\to n}$ and probability distribution divergence function L, we define an **optimal fit functor** $O_{P_{k\to n}L}$ between $Dist_n$ and $Vect_k$ to be a map of the following form:

- *Objects*: The object $D \in Dist_n$ is mapped to an **Lp-approximation** object $v \in Vect_k$ such that $L(D, P_{k\to n}(v))$ is minimized.
- *Arrows*: The arrow $(D_1, D_2, F) \in Dist_n$ that maps D_1 to D_2 is mapped to the unique arrow in $Vect_k$ that maps $O_{P_{k\to n}L}(D_1)$ to $O_{P_{k\to n}L}(D_2)$.

Lemma 3.1. *For any simple parametric model $P_{k\to n}$ and probability distribution divergence L, any optimal fit functor $O_{P_{k\to n}L}$ is a functor.*

Proof: First, since both $Vect_k$ and $Dist_n$ are thin, $O_{P_{k\to n}L}$ must preserve the identity.
Next, consider the distribution maps (D_1, D_2, F_{12}) and (D_2, D_3, F_{23}). By definition, $O_{P_{k\to n}L}(D_2, D_3, F_{23}) \circ O_{P_{k\to n}L}(D_1, D_2, F_{12})$ is the sum $u_{12}+u_{23}$ such that $O_{P_{k\to n}L}D_1 + u_{12} = O_{P_{k\to n}L}D_2$ and $O_{P_{k\to n}L}D_2 + u_{23} = O_{P_{k\to n}L}D_3$. This implies that $O_{P_{k\to n}L}D_1 + u_{12} + u_{23} = O_{P_{k\to n}L}D_3$, so $u_{12}+u_{23}$ is the unique arrow mapping $O_{P_{k\to n}L}D_1$ to $O_{P_{k\to n}L}D_3$. Therefore $O_{P_{k\to n}L}(D_2, D_3, F_{23}) \circ O_{P_{k\to n}L}(D_1, D_2, F_{12}) = O_{P_{k\to n}L}((D_2, D_3, F_{23}) \circ (D_1, D_2, F_{12}))$, so $O_{P_{k\to n}L}$ preserves composition. ■

- **Example:** Consider the case where D in $Dist_1$ is the distribution of rolls of a six-sided die with the following weights:

Roll	Weight
1	0.05
2	0.15
3	0.3
4	0.3
5	0.15
6	0.05

  If $P_{2\to 1}$ is a function that accepts the vector (μ, θ) and returns a univariate normal distribution with mean μ and variance θ, and L is KL-divergence, then $O_{P_{2\to 1}L}D \approx (1.5, 1.45)$ (the equivalent to fitting a normal distribution to an infinite number of samples from D).

For some continuous parametric model $P_{k\to n}$, we now define the **generative functor** $G_{P_{k\to n}}$ to be a map between $Vect_k$ and $Dist_{n_c}$ of the following form:

- *Objects*: The object $v \in Vect_k$ is mapped to $P_{k\to n}(v)$.
- *Arrows*: The unique arrow between v_1 and v_2 is mapped to the distribution map $(P_{k\to n}(v_1), P_{k\to n}(v_2), F)$ where F is the set of all functions f such that $d \sim P_{k\to n}(v_1)$ implies $f(d) \sim P_{k\to n}(v_2)$.

Lemma 3.2. *For any parametric model $P_{k\to n}$, $G_{P_{k\to n}}$ is a functor.*

Proof: First, since both $Vect_k$ and $Dist_{nc}$ are thin, $G_{P_{k\to n}}$ must preserve the identity.
Next, by the definition of a continuous parametric model, for any $v_1, v_2 \in Vect_k$, $P_{k\to n}(v_1)$ and $P_{k\to n}(v_2)$ are continuous probability distributions. By Lemma 2.2, this implies that there must be an arrow between $P_{k\to n}(v_1)$ and $P_{k\to n}(v_2)$. Since both $Vect_k$ and $Dist_{n_c}$ are thin, this implies that $G_{P_{k\to n}}$ must preserve composition. ■

- **Example:** If we define $P_{6\to 2}$ as a function that accepts the vector $(\mu_1, \mu_2, \theta_{11}, \theta_{12}, \theta_{21}, \theta_{22})$ and returns a bivariate normal distribution with mean and covariance matrix:

$$\mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} \Sigma = \begin{pmatrix} \theta_{11} & \theta_{12} \\ \theta_{21} & \theta_{22} \end{pmatrix}$$

  Then $G_{P_{6\to 2}}$ maps the vector $(0, 0, 1, 0, 0, 1)$ to the standard bivariate normal distribution.

We can also define a **continuous optimal fit functor** $O_{P_{k\to n}L_c}$ to be a functor of the same form as $O_{P_{k\to n}L}$ that maps $Dist_{n_c}$ to $Vect_k$. By the same argument as Lemma 3.1, $O_{P_{k\to n}L_c}$ is a functor.

- **Example:** Let's consider the class of three-layer fully-connected neural networks with sigmoid activations and layer sizes (784, 784, 784). We can define an instance of such a network with the weight vector $v = (\theta_1, \theta_2, ..., \theta_{1229312})$. Furthermore, we can use this network to define a probability distribution over R^{784} by applying this network to samples drawn from the multivariate standard normal distribution over R^{784}. Now let's define $P_{1229312\to 784}$ to be the function that maps a 1229312-element vector v to the distribution defined by the neural network with weight vector v. Then we see that:
  - Say D is the distribution in $Dist_{784_c}$ of (28x28) pictures of digits. The functor $O_{P_{1229312\to 784}L_c}$ will map D to the vector $v \in Vect_{1229312}$ such that the neural network that v parameterizes is the best approximation of the digit picture distribution according to L.
  - The functor $G_{P_{1229312\to 784}}$ maps $v \in Vect_{1229312}$ to the probability distribution in R^{784} defined by the neural network parameterized by v.

3.2 Optimal Model Fitting Adjunctions and Monad

Lemma 3.3. *For any parametric model $P_{k\to n}$, probability distribution divergence L, and continuous optimal fit functor $O_{P_{k\to n}L_c}$, the generative functor $G_{P_{k\to n}}$ is the left adjoint of $O_{P_{k\to n}L_c}$.*

Proof: First, we define the unit $\eta :: id_{Vect_k} \to O_{P_{k\to n}L_c} \circ G_{P_{k\to n}}$. Since $L(D_1, D_2) = 0$ iff $D_1 = D_2$, we see that $(O_{P_{k\to n}L_c} \circ G_{P_{k\to n}})v = O_{P_{k\to n}L_c}\ P_{k\to n}(v) = v$. Therefore, since $Vect_k$ is thin, the component η_v must be the identity arrow so η is natural.
Next, we define the counit $\epsilon :: G_{P_{k\to n}} \circ O_{P_{k\to n}L_c} \to id_{Dist_{n_c}}$. Since $(G_{P_{k\to n}} \circ O_{P_{k\to n}L_c})D = v$, where v is D's L_P-approximation in $Vect_k$, the component $\epsilon_{P_{k\to n}(v)}$ must map from $P_{k\to n}(v)$ to D. By Lemma 2.2, this arrow must exist. Since $Dist_{n_c}$ is thin, this implies that ϵ is natural. ■

Lemma 3.4. *For any parametric model $P_{k\to n}$, probability distribution divergence L, and continuous optimal fit functor $O_{P_{k\to n}L_c}$, the generative functor $G_{P_{k\to n}}$ is the right adjoint of $O_{P_{k\to n}L_c}$.*

Proof: First, we define the unit $\eta :: id_{Dist_{n_c}} \to G_{P_{k\to n}} \circ O_{P_{k\to n}L_c}$. Since $Dist_{n_c}$ is thin and connected, η_D must be the single arrow from D to $P(O_{P_{k\to n}L_c}D)$ and η must be natural.
Next, we define the counit $\epsilon :: O_{P_{k\to n}L_c} \circ G_{P_{k\to n}} \to id_{Vect_k}$. Since $(O_{P_{k\to n}L_c} \circ G_{P_{k\to n}})v = v$, the component ϵ_v must be the identity arrow. Therefore ϵ is natural. ■

We can now define the model fitting monad $M_{P_{k\to n}L_c}$ based on the adjunction between some $O_{P_{k\to n}L_c}$ and $G_{P_{k\to n}}$ defined in Lemma 3.4. The η of $M_{P_{k\to n}L_c}$ is the unit of this adjunction, so η maps some distribution $D \in Dist_n$ to its best L-approximation $P_{k\to n}(v)$.
Similarly, the μ of $M_{P_{k\to n}L_c}$ is $G_{P_{k\to n}}\epsilon O_{P_{k\to n}L_c}$, where ϵ is the counit of the adjunction. Since ϵ is the identity, μ represents the idempotence of distribution approximation according to a non-random probability divergence (i.e. approx(approx(dist)) = approx(dist)).

- **Example:** Building on the digit approximation example, the η of $M_{P_{k\to n}L_c}$ is an arrow that maps the distribution of digit images D to D_A, the best approximation of D such that there exists some three-layer fully-connected neural network with sigmoid activations and layer sizes $(784, 784, 784)$ that defines D_A.

4 TRAINING ON DATA

In this section we morph the previous section's concept of "optimal approximation" into the process of drawing samples from a distribution (selecting a training set) and training a model on those samples.

4.1 Training Functor

Now, for some parametric model $P_{k\to n}$ and **training set** of m training samples $S = \{s_1, s_2, ..., s_m\}$ drawn from the uniform distribution, we define a **training functor** $T_{P_{k\to n}S}$ between $Dist_{n_c}$ and $Vect_k$ to be a map of the following form:

- *Objects*: The object $D \in Dist_{n_c}$ is mapped to an **S-approximation** object $v \in Vect_k$ such that for some function element f of (U, D, F), the distribution $P_{k\to n}(v)$ maximizes the likelihood (over all distributions in the image of $P_{k\to n}$) of the set of samples $\{f(s_1), f(s_2), ..., f(s_m)\}$.
- *Arrows*: The arrow $(D_1, D_2, F_{12}) \in Dist_n$ that maps D_1 to D_2 is mapped to the unique arrow in $Vect_k$ that maps $T_{P_{k\to n}S}(D_1)$ to $T_{P_{k\to n}S}(D_2)$.

By the same argument as Lemma 3.1, $T_{P_{k\to n}S}$ is a functor.

4.2 Training Functors Converge to the Continuous Optimal Fit Functor

We can now describe the relationship between the training functors and the continuous optimal fit functor. Let's first define a **functor distribution** to be a random variable over a set of functors mapping $Dist_{n_c}$ to $Vect_n$. Next, for any integer M let's define the **size-M training functor distribution** T_M to be the uniformly distributed random variable over the set of all training functors $T_{P_{k\to n}S}$ such that the training set S is size M.

Now for some continuous probability distribution $D \in Dist_{n_c}$ and continuous parametric model $P_{n\to k}$, let's define the **expected KL-Divergence** of the functor distribution F to be $E_{f\sim F}[KL(D, P_{n\to k}(f(D)))]$. That is, the expected KL-Divergence measures how well the training functors in $T_{P_{k\to n}S}$ learn the distribution D.

We can now define the **functor distribution comparison category** $F_D P_{n\to k}$ to be the ordering over all functor distributions such that there is an arrow mapping functor distribution F_1 to functor distribution F_2 if the expected KL-Divergence of F_1 is less than or equal to that of F_2. Based on this, let's define the **optimal KL-Divergence** to be $KL(D, P_{n\to k}(O_{P_{n\to k}KL_c}(D)))$ and let's note that by the definition of $O_{P_{n\to k}KL_c}$, the terminal object in $F_D P_{n\to k}$ must have expected KL-Divergence no smaller than the optimal KL-Divergence.

Next, let's define the **training functor distribution comparison subcategory** $Tr_D P_{n\to k}$ to be the subcategory of $F_D P_{n\to k}$ formed

from just the training functor distributions. Now let's consider $lim\ I$, where I is the functor between $Tr_D P_{n\to k}$ and $F_D P_{n\to k}$ that acts as the identity on objects and morphisms.

Lemma 4.1. *For some continuous probability distribution $D \in Dist_{n_c}$ and continuous parametric model $P_{n\to k}$, the limit of the functor I that embeds the training functor distribution comparison subcategory $Tr_D P_{n\to k}$ into the functor distribution comparison category $F_D P_{n\to k}$ is a functor distribution with expected KL-divergence equal to the optimal KL-Divergence.*

Proof:
Assume for contradiction that the statement is not true. Then $lim\ I$ is some functor distribution $F_D P_{n\to k}$ such that the expected KL-divergence of $lim\ I$ is not the optimal KL-Divergence. By the definition of optimal KL-Divergence, the following must hold:

$$\epsilon = E_{f\sim lim\ I}[KL(D, P_{n\to k}(f(D)))] - KL(D, P_{n\to k}(O_{P_{n\to k}KL_c}(D))) \geq 0$$

Now we can choose an M that is large enough such that with arbitrarily large probability we can draw $s_1, s_2, ..., s_M \sim U$ and choose some function element g_D of (U, D, F) to get $g_D(s_1), g_D(s_2), ..., g_D(s_M) = x_1, x_2, ..., x_M \sim D$ such that for any probability function Pr:

$$\left| \int^x Pr_D(x) log\left(Pr(x)\right)\ dx - \sum_{i=1}^{M} log\left(Pr(x_i)\right) \right| < \frac{\epsilon}{4}$$

Now let's set $S = s_1, s_2, ..., s_M$ and define Pr_{T_S} to be the probability function $P_{n\to k}(T_{P_{k\to n}S}(D))$. Let's also define Pr_D to be the probability function of D and Pr_{opt} to be the probability function of $P_{n\to k}(O_{P_{n\to k}KL_c}(D))$. Now since Pr_{T_S} maximizes $f_1(Pr) = \sum_{i=1}^{M} log\left(Pr(x_i)\right)$ and Pr_{opt} maximizes $f_2(Pr) = \int^x Pr_D(x) log\left(Pr(x)\right)\ dx$, we have that with arbitrarily large probability:

$$\left(\int^x Pr_D(x) log\left(Pr_{opt}(x)\right)\ dx - \int^x Pr_D(x) log\left(Pr_{T_S}(x)\right)\ dx \right) < \frac{\epsilon}{2}$$

Therefore, for the size-M training functor distribution T_M, we can see that without any randomness:

$$E_{T_{P_{k\to n}S}\sim T_M}\left[KL(D, P_{n\to k}(T_{P_{k\to n}S}(D)))\right] - KL(D, P_{n\to k}(O_{P_{n\to k}KL_c}(D))) =$$
$$E_{T_{P_{k\to n}S}\sim T_M}\left[\int^x Pr_D(x)\ log\left(\frac{Pr_D(x)}{Pr_{T_S}(x)}\right)\ dx - \int^x Pr_D(x)\ log\left(\frac{Pr_D(x)}{Pr_{opt}(x)}\right)\ dx \right] =$$
$$E_{T_{P_{k\to n}S}\sim T_M}\left[\int^x Pr_D(x) log\left(Pr_{opt}(x)\right)\ dx - \int^x Pr_D(x) log\left(Pr_{T_S}(x)\right)\ dx \right] < \epsilon$$

This implies that there is no arrow mapping $lim\ I$ to T_M, so we have reached a contradiction.

4.3 Model Compression Functor

Given a set of m training samples $S = \{s_1, s_2, ..., s_m\}$ drawn from the uniform distribution and two continuous parametric models $P_{k_1\to n}$, $P^*_{k_2\to n}$ such that the image of $P^*_{k_2\to n}$ over R^{k_2} is a subset of the image of $P_{k_1\to n}$ over R^{k_1}, we can define an **optimal model compression functor** $C_{P^*_{k_2\to n}S_{P_{k_1\to n}}}$ between $Vect_{k_1}$ and $Vect_{k_2}$ to be a map of the following form:

- *Objects*: The object $v \in Vect_{k_1}$ is mapped to the $u \in Vect_{k_2}$ such that for some function element f of $(U, P_{k_1\to n}(v), F)$, the distribution $P^*_{k_2\to n}(u)$ maximizes the likelihood (over all distributions in the image of $P^*_{k_2\to n}$) of the set of samples $\{f(s_1), f(s_2), ..., f(s_m)\}$.
- *Arrows*: The unique arrow between v_1 and v_2 in $Vect_{k_1}$ is mapped to the unique arrow between $C_{P^*_{k_2\to n}S_{P_{k_1\to n}}}v_1$ and $C_{P^*_{k_2\to n}S_{P_{k_1\to n}}}v_2$ in $Vect_{k_2}$.

Lemma 4.2. *For any set of m training samples $S = \{s_1, s_2, ..., s_m\}$ drawn from the uniform distribution and two continuous parametric models $P_{k_1\to n}$, $P^*_{k_2\to n}$ such that the image of $P^*_{k_2\to n}$ over R^{k_2} is a subset of the image of $P_{k_1\to n}$ over R^{k_1}, any optimal model compression functor $C_{P^*_{k_2\to n}S_{P_{k_1\to n}}}$ is a functor.*

Proof: First, since both $Vect_{k_1}$ and $Vect_{k_2}$ are thin, $C_{P^*_{k_2\to n}S_{P_{k_1\to n}}}$ must preserve the identity.
Next, consider the objects $v_1, v_2, v_3 \in Vect_{k_1}$ and the arrows $u_{12}, u_{23} \in Vect_{k_1}$ such that $v_1 + u_{12} = v_2$, $v_2 + u_{23} = v_3$ and $v_1 + u_{12} + u_{23} = v_3$.

(1) $C_{P^*_{k_2\to n}S_{P_{k_1\to n}}}$ maps the arrow u_{12} to the arrow between $C_{P^*_{k_2\to n}S_{P_{k_1\to n}}}v_1$ and $C_{P^*_{k_2\to n}S_{P_{k_1\to n}}}v_2$.
(2) $C_{P^*_{k_2\to n}S_{P_{k_1\to n}}}$ maps the arrow u_{23} to the arrow between $C_{P^*_{k_2\to n}S_{P_{k_1\to n}}}v_2$ and $C_{P^*_{k_2\to n}S_{P_{k_1\to n}}}v_3$.
(3) $C_{P^*_{k_2\to n}S_{P_{k_1\to n}}}$ maps the arrow $u_{12} + u_{23}$ to the arrow between $C_{P^*_{k_2\to n}S_{P_{k_1\to n}}}v_1$ and $C_{P^*_{k_2\to n}S_{P_{k_1\to n}}}v_3$.

By Lemma 2.4 we know all three of these arrows exist in $Vect_{k_2}$ and are unique, so $C_{P^*_{k_2\to n}S_{P_{k_1\to n}}}(u_{12} + u_{23}) = C_{P^*_{k_2\to n}S_{P_{k_1\to n}}}u_{12} + C_{P^*_{k_2\to n}S_{P_{k_1\to n}}}u_{23}$ and $C_{P^*_{k_2\to n}S_{P_{k_1\to n}}}$ preserves composition. ∎

It is not generally the case that $C_{P^*_{k_2 \to n} S_{P_{k_1 \to n}}} \circ T_{P_{k_1 \to n} S} = T_{P^*_{k_2 \to n} S}$. That is, the model compression procedure may produce a different set of parameters for $P^*_{k_2 \to n}$ than would $T_{P^*_{k_2 \to n} S}$. In certain circumstances the model compression approach can yield better generalization performance than training the smaller model directly [1].

- **Example:** To continue the digit approximation theme, consider the case where we have constructed two three-layer fully connected neural networks: network $P^A_{1229312 \to 784}$ with layer sizes $(784, 784, 784)$ and network $P^B_{156800 \to 784}$ with layer sizes $(784, 100, 784)$. Then a training functor $T_{P^A_{1229312 \to 784} S}$ will map the distribution of digit images D to a parameter vector v_a in $R^{1229312}$. The model compression functor $C_{P^B_{156800 \to 784} S_{P^A_{1229312 \to 784}}}$ can then map v_a to the parameter vector v_b in R^{156800} such that $P^B_{156800 \to 784}(v_b)$ is a good approximation for $P^A_{1229312 \to 784}(v_a)$. Note that v_b is not necessarily equal to $T_{P^B_{156800 \to 784} S} D$.

4.4 Optimization (TODO)

- **TODO:** The major benefit of our existing structure in Vect where parameters are related based on "update" addition vectors is that it provides a natural framework to discuss optimization. How can represent this?
- **TODO:** Can we add an additional degree of freedom to the training functor where we don't maximize the likelihood but instead have the training functor perform the distribution to vector mapping with some sort of optimization approach (such as some kind of optimization surface traversal like gradient descent)?
- **TODO:** Can we change the definition of the training functor to instead map to a local minima? If we do this, can we define the global minima to be some sort of categorical limit/colimit based on arrows that map between the global and local minima? Or perhaps we can define an ordering on the training functors and have the global minima functor be the terminal object?

5 ALTERNATIVE FORMULATION

- Our definition of distribution map composition feels unnatural since it relies on the uniqueness of the distribution map between two distributions, rather than a pointwise definition. Unfortunately, the more natural definition $(B, C, G) \circ (A, B, F) = (A, C, \{g \circ f | \forall g \in G, \forall f \in F\})$ is not closed under composition since there may be functions h such that $a \in A$ implies $h(a) \in C$ that cannot be expressed as the composition of some $g \in G$ and $f \in F$. One alternative formulation would be to instead describe $Dist_{n_c}$ as the category where objects are distributions and the arrows between objects D_a and D_b are functions that can be expressed as $f_{D_a} g_{D^*_1} ... f_{D^*_m} g_{D_b}$ where f_D and g_D are the functions described in Lemma 2.2. However, this modification would make $Dist_{n_c}$ no longer thin, so the adjunctions and monad that we introduce in Section 3.2 would no longer exist unless we modify $Vect_n$ as well.

REFERENCES

[1] Hinton, G., Vinyals, O., and Dean, J. (2015). Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531.*