Evaluation Methods for Generative Models

Honours Electrical Engineering Bachelor's Thesis

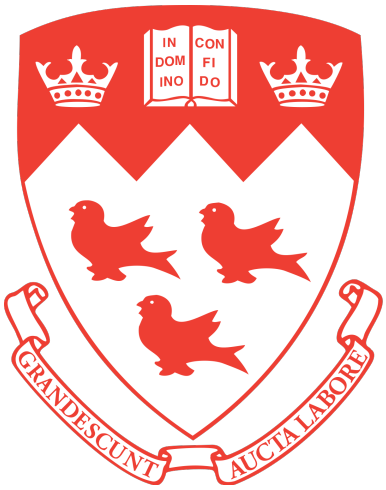
Anja Kroon
260886624
anja.kroon@mail.mcgill.ca

ECSE 478
McGill University

Supervisor:
Professor Mark Coates
mark.coates@mcgill.ca

This work is in conjunction with
PhD candidate Florence Regol.

April 2023



1 Abstract

Generative models, such as ChatGPT and DALL-E, are playing an increasingly important role in our society transforming many fields such as education, business, and art. As we begin to rely upon and trust the outputs of generative models, accuracy becomes increasingly important. Existing evaluation metrics for generative models compare the ground truth distribution, q, to the learned distribution, p, but are approximate, insensitive to specific failure types, or are task-specific. With improved evaluation metrics, better models can be developed. This work proposes an evaluation metric for categorical generative models with statistical guarantees in high dimensional probability space settings $\approx \Omega \geq 10^6$ which is generalizable, robust to mismatches between p and q and provides interpretative results. For experimentation, we return to the synthetic data setting where the ground truth distribution p is known. The datatype is categorical meaning there is a finite number of categories. The proposed evaluation method for categorical generative models involves successively binning a large probability space into smaller probability spaces such that statistical tests can be applied enabling sample complexity and statistical significance considerations.

The objectives of this project as outlined in the project proposal are to contribute to the development of an evaluation method for categorical generative models. My tasks were to contribute to the experimental codebase, contribute to documentation (writing), and produce experimental results. These three objectives were achieved through the guidance and help of PhD candidate Florence Regol and Professor Mark Coates. For a comprehensive list of learning outcomes, see section 8.

2 Acknowledgements

Through the honours thesis, I have worked closely with Florence Regol, the primary investigator of this research, and Professor Coates. None of my work would have been possible without their patience and support. I am humbled and thankful for the opportunity to help with this research having learned a tremendous amount. I would like to specifically thank Florence for answering my numerous questions and pointing me in the direction of resources to learn further. I would also like to thank Professor Coates for providing me this opportunity and allowing me to be a member of the Networks Research Laboratory this past year.

Contents

3 Introduction

Generative models. Generative models are a class of machine learning models that aim to learn the underlying distribution, p, of a given dataset and then generate new data samples from learned distribution, q. These models have the potential to revolutionize many fields such as healthcare in generating protein sequences for drug discovery and design [1]. Generative models can also be used to create personalized learning materials transforming education [2]. Moreover, generative models have enabled breakthroughs in natural language processing, such as the development of language models that can generate coherent and fluent text, leading to the development of chatbots and other conversational AI systems such as ChatGPT [3]. The evaluation metrics for generative models are an important consideration as they help researchers distinguish the highest-performing models to producing the most accurate samples closely representing the ground truth distribution. As generative models are increasingly trusted by the public, having correct, robust, and accurate evaluation metrics is of utmost importance.

Existing Evaluation Metrics. Classically, generative models are evaluated upon their log-likelihood (LL). For a random sample $x_1, \cdots, x_n$ the LL is the product of probability density functions, $L(\theta) = f(x_1;\theta)f(x_2;\theta)\cdots f(x_n;\theta)$. The LL can thus be defined as,

$$LL(\theta|X) = log(L(\theta|X)) \tag{1}$$

where LL represents the log-likelihood function, θ represents the parameters of the statistical model, and X represents the observed data. The LL is a commonly used metric because it evaluates how likely the generated samples are to come from p, the ground truth distribution. While the LL provides an interpretable measure of performance and is widely applicable across many types of generative models, a good LL is not necessarily indicative of good sample generation [4].

Further, sometimes only the upper bound of the LL is accessible. The upper bound of the LL does not provide a clear evaluation of generative model performance. For this reason, task-oriented metrics were created to evaluate the quality of the generated samples. One such metric is the Fréchet Inception Distance (FID) which measures the distance between the feature representations of the generated images and those of the real images [5]. Task-oriented metrics are a good evaluation method but constrain the datatype making it applicable to only a few fields.

In an attempt to create a generalizable metric, a new class of evaluation metrics was created – coverage metrics. Intuitively, these metrics measure what percentage of p is covered by q and vice versa. The limitation of this approach, however, is that they require there to be a meaningful notion of distance in the distribution space and thus require a specific sample ordering. For the categorical data of consideration, the coverage metrics are thus only applicable when the data has explicitly been transformed. Due to the aforementioned limitations of the LL, task-oriented metrics and coverage metrics, a new evaluation metric must be developed.

Proposed method. The proposed evaluation method for categorical generative models seeks to address the limitations of existing evaluation methods. It successively bins a large probability space into smaller probability spaces figure 1. On the inherently smaller binned versions of p and q, statistical tests become applicable providing a venue for statistical considerations. This now enables a metric called the total variation error, d_{TV}, which measures the statistical distance between two distributions.

$$d_{TV} = \frac{1}{2}||\mathbf{p} - \mathbf{q}|| \tag{2}$$

Models which produce a smaller d_{TV} error more closely follow the original distribution, p, and perform better. It is notable that the binning process introduces error as it simplifies the larger

probability space into smaller ones. This, however, is surmountable as a poor model which struggles to correctly assign probability mass function (pmf) values in a larger probability space will necessarily also struggle to correctly assign pmf values in a smaller space. Intuitively, the metric is able to provide a ranking between generative models and an interpretable metric which is within a measurable amount to the true error.

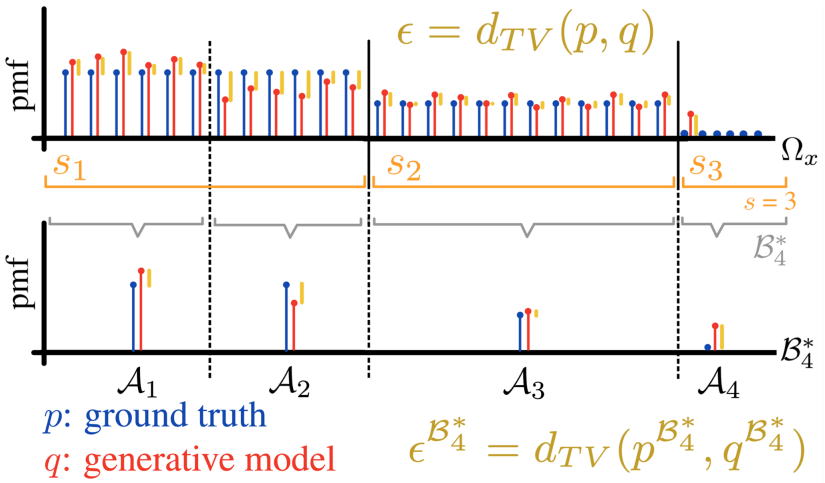


Figure 1: Overview of the binning procedure from [6]. The initial distribution p has $s = 3$ flat regions on the $|\Omega|$ space contained in $\mathbf{S} = \{\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3\}$. The binning procedure with k=4 decides to split $\mathcal{S}_1 = \{\mathcal{A}_1, \mathcal{A}_2\}$ to form $\mathcal{B}_4^* = \{\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \mathcal{A}_4\}$. The (*) indicates statistical significance. With this final result, $d_{TV}(p^{\mathcal{B}_4}, q^{\mathcal{B}_4})$ is maximized.

Metric Qualities. The proposed metric considers sample complexity, and statistical significance, is applicable in large sample spaces $\Omega \geq 10^6$, is robust to different patterns of mismatch between p and q, and provides interpretable results. Sample complexity refers to the number of training examples needed to achieve a certain level of performance having significant implications in feasibility. Even with access to p, the number of samples m needed to estimate the d_{TV} in a sample space Ω with some error ϵ with probability $1-\delta$ would be of order $m \propto \sqrt{|\Omega|}$ [7]. Statistical significance is vital and has largely been ignored in previous works. Statistical significance allows us to assess whether the differences between the generated samples and the ground truth samples are meaningful or due to random chance. Creating a metric applicable in large sample spaces is an important consideration as complex tasks are of higher interest with stronger potential benefits to society. Different patterns of mismatch between p and q can unjustly rank models lower than others. Thus, the metric should also be robust to this type of scenario.

Synthetic Data Experimental Setting. For experimentation, we first return to a synthetic setting where p is known. This enables us to determine directly whether or not the model assigns the correct pmf value to each element in the sample space, Ω. In a synthetic setting, the target distribution is crafted to resemble real-world data.

Real Data Experimental Setting. Next, we aim to transition to real data to demonstrate the proposed evaluation metric capability of operating in realistic high-dimensional settings. For this task, we consider protein sequences from Proteinnet7 as the categorical data [8]. Proteinnet7 is a publicly available benchmark dataset designed for evaluating the performance of machine learning models in predicting the physical properties of proteins. It consists of a set of millions protein structures of various lengths with their amino acid sequences. The dataset exhibits categorical behavior with certain strands reappearing frequently. Experimentation with this dataset is currently

underway and will thus not be reported upon in this thesis.

Key Contributions. A proposed new evaluation method for categorical generative models is introduced. It involves successively binning a high dimensional probability space into smaller probability spaces such that statistical tests may be applied. The approach provides sample complexity and statistical significance considerations, is applicable in large sample spaces $\Omega \geq 10^6$ and is robust to different patterns of mismatch between p and q and provides interpretable results.

4 Methodology

4.1 Problem Setting

Generative models attempt to learn a ground truth distribution p and produce samples from the learned distribution q. For the same task, various generative models may be developed leading to multiple learned distributions q_1, q_2,... q_n. It is our objective to distinguish which generative model performs best and provide a ranking. The best generative model learns a distribution q closest to the original distribution p. From the statistical identity testing problem in statistical distribution testing, we can formalize the closeness of p and q on a discrete sample space Ω as the total variation distance,

$$d_{TV}(p,q) \triangleq \frac{1}{2}||p-q||_1 = \frac{1}{2}\sum_{x\in\Omega}|p_x - q_x| \tag{3}$$

Here, p_x refers to the probability mass function of x, $p(x)$. We have access to the ground truth distribution q and sample access to the learned distribution q. The test declares the distributions are d_{TV} close with a probability of at least $1-\delta$. Typically, identity testing cannot be applied to evaluate p and q from generative models as there are not enough samples, m, from the very large original sample space $|\Omega|$. Particularly, the provably most powerful test states, $m \propto \sqrt{|\Omega|}$ [9]. The proposed evaluation method seeks to successively bin the probability space into smaller probability space such that the d_{TV} metric can be applied on the binned spaces. d_{TV} is a preferred evaluation approach as it provides statistical guarantees and an intuitive ranking metric.

4.2 Preliminaries

The set Ω is partitioned into smaller sets $\rho(\Omega) = \{\{\mathcal{A}_1, \dots\} | \cup_i \mathcal{A}_i = \Omega, \mathcal{A}_j \cap \mathcal{A}_i = \emptyset \, \forall i \neq j\}$. A specific partitioning is defined as, $\mathcal{B} = \{\mathcal{A}_1, \mathcal{A}_2 \dots\}$. This partitioning introduces several smaller probability distributions which given our definitions can be defined as, $\rho^k(\Omega) = \{\mathcal{B} \in \rho(\Omega), |\mathcal{B}| = k\}$. Thus, k refers to the number of small distributions which are defined as, $\mathcal{B}^k \in \rho^k(\Omega)$. The distribution p will induce a binned distribution $p^{\mathcal{B}^k}$ defined as,

$$p^{\mathcal{B}^k}_{\mathcal{A}_i} \triangleq \sum_{x\in\mathcal{A}_i} p_x \,, \tag{4}$$

Next, we apply the triangle inequality to relate the binned and non-binned versions in terms of the total variation error, d_{TV}. First, the triangle inequality is stated followed by a lemma relating the two qualities.

Triangle Inequality: Establish, $|x| = \max\{x, -x\} and \pm x \leq |x|$. Then,

$$\begin{aligned} a + b \leq |a| + b \leq |a| + |b|, \quad \text{and} \\ -a - b \leq |a| - b \leq |a| + |b|. \end{aligned}$$

Lemma 1: $\mathcal{B} \in \rho(\Omega) \implies d_{TV}(p^{\mathcal{B}}, q^{\mathcal{B}}) \leq d_{TV}(p, q)$

$$
\begin{aligned}
d_{TV}(p^{\mathcal{B}}, q^{\mathcal{B}}) &= \frac{1}{2} \sum_{\mathcal{A}_i \in \mathcal{B}} |p^{\mathcal{B}}_{\mathcal{A}_i} - q^{\mathcal{B}}_{\mathcal{A}_i}| && \text{by equations 3 and 4} \\
&= \frac{1}{2} \sum_{\mathcal{A}_i \in \mathcal{B}} |\sum_{x \in \mathcal{A}_i} p_x - \sum_{x \in \mathcal{A}_i} q_x| && \text{by definition of } \mathcal{B} \\
&\leq \frac{1}{2} \sum_{\mathcal{A}_i \in \mathcal{B}} \sum_{x \in \mathcal{A}_i} |p_x - q_x| && \text{by definition of the triangle inequality} \\
&= \frac{1}{2} \sum_{x \in \Omega} |p_x - q_x| && \text{by definition of } \mathcal{A}_i \\
d_{TV}(p^{\mathcal{B}}, q^{\mathcal{B}}) &\leq d_{TV}(p, q)
\end{aligned}
$$

Thus, the total variation error on the binned distribution will always be less than the actual, non-binned total variation error. When evaluating a model, this implies that a high d_{TV} value, indicative of a poor-performing generative model, on the binned distributions necessarily implies a high d_{TV} value on the actual, non-binned distributions.

We can expand the implications of the above conclusion by imposing the constraint that each newly constructed binning is a partition of the previous binning ($\mathcal{B}^{i-1} \in \rho(\mathcal{B}^i)$). This requires building a sequence of probability spaces where the first binning, $\mathcal{B}^k \in \rho(\Omega)$, constitutes the probability space of the entire sample space. This constraint enables us to conclude that the total variation error, d_{TV}, is increasing over the binnings $\mathcal{B}_1$ to $\mathcal{B}_k$.

$$d_{TV}(p^{\mathcal{B}^{i-1}}, q^{\mathcal{B}^{i-1}}) \leq d_{TV}(p^{\mathcal{B}^i}, q^{\mathcal{B}^i}) \cdots \leq d_{TV}(p, q) \tag{5}$$

This also implies that the size of each partition will also increase over the bins,

$$|\mathcal{B}^1| \leq |\mathcal{B}^2| \cdots \leq |\mathcal{B}^k| \tag{6}$$

Equation 6 holds if Lemma 1 holds for each step.

4.3 Proposed Evaluation Method

Intuitively, the proposed methodology states we can compare different generative models with their d_{TV} values over their binned distributions at the same granularity level, i. Say we have two generative models tasked to learn the same underlying distribution, p. Model 1 and model 2 produce two learned distributions q_i and q_2, respectively. Now, we can compare the performance of these generative models by, $d_{TV}(p^{\mathcal{B}^i}, q_1^{\mathcal{B}^i})$ vs. $d_{TV}(p^{\mathcal{B}^i}, q_2^{\mathcal{B}^i})$, declaring the model with the lower error the better model. We can additionally opt to declare an error threshold, ϵ_{thresh}, where models can be distinguished as either passing the test threshold or not passing the test threshold. It is important to note that a declaration of $d_{TV}(p^{\mathcal{B}^i}, q_1^{\mathcal{B}^i}) \leq d_{TV}(p^{\mathcal{B}^i}, q_2^{\mathcal{B}^i})$ for example does not imply $d_{TV}(p, q_1) \leq d_{TV}(p, q_2)$. However, the binned comparisons capture general trends, thus proving useful as a comparison metric.

4.4 Binning

The binned version of the distributions, $p^{\mathcal{B}^i}$, $q^{\mathcal{B}^i}$, should follow as closely as possible p, q in order to promote the most accurate comparison. Within binning, the constraints are imposed that all binnings are partitions of the sample space Ω, $\{\mathcal{B}^1, \ldots \mathcal{B}^k\} \in \rho(\Omega)$. Secondly, we impose the

constraint that each bin is a partition of the preceding bin, $\mathcal{B}^{i-1} \in \rho(\mathcal{B}^i)$. The binning procedure is first described with a toy example:

Toy Binning Example: We commence by constructing the first partition $\mathcal{B}^1$

$$\Omega = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\} \tag{7}$$

$$\Omega = \overbrace{\{\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}\}}^{R} \qquad \Delta = 2 \quad p_{max} = 10 \tag{8}$$

$$\Omega = \overbrace{\{\{1, 2, 3, 4, 5, 6, 7\}}^{R}, \overbrace{\{8, 9, 10\}}^{\mathcal{A}^1}\} \qquad \Delta = 2 \quad p_{max} = 7 \tag{9}$$

$$\Omega = \overbrace{\{\{1, 2, 3, 4\}}^{R}, \overbrace{\{5, 6, 7\}}^{\mathcal{A}^2}, \overbrace{\{8, 9, 10\}}^{\mathcal{A}^1}\} \qquad \Delta = 2 \quad p_{max} = 4 \tag{10}$$

$$\Omega = \{\overbrace{\{1\}}^{R}, \overbrace{\{2, 3, 4\}}^{\mathcal{A}^3}, \overbrace{\{5, 6, 7\}}^{\mathcal{A}^2}, \overbrace{\{8, 9, 10\}}^{\mathcal{A}^1}\} \qquad \Delta = 2 \quad p_{max} = 1 \tag{11}$$

$$\mathcal{B}^1 = \{\overbrace{\{1\}}^{\mathcal{A}^4}, \overbrace{\{2, 3, 4\}}^{\mathcal{A}^3}, \overbrace{\{5, 6, 7\}}^{\mathcal{A}^2}, \overbrace{\{8, 9, 10\}}^{\mathcal{A}^1}\} \qquad \Delta = 2 \quad p_{max} = 1 \tag{12}$$

In the toy example, the numerical values are arbitrary to demonstrate the binning procedure. As the binning occurs over a probability mass function, no value will be greater than 1 or less than 0. A tolerance parameter Δ is set. Δ will ultimately represent the maximum difference between any two elements in the same bin. The choice of Δ is directly tied to the number of partitions $|\mathcal{B}|$ the final iteration will include. In the binning procedure, a remainder, R, first includes all elements of Ω. The maximum probability mass, p_{max}, is identified. Elements are grouped together into a first partition, $\mathcal{A}^1$ such that $p_x \geq p_{max} - \Delta$. After all elements have been grouped, R is redefined with the remaining elements. Once again, p_{max} is identified in R. Elements are grouped into $\mathcal{A}^2$ such that $p_x \geq p_{max} - \Delta$ as before. This process continues iteratively until all elements have been grouped into partitions and the $R = \emptyset$.

We then construct the following $\mathcal{B}^2, \ldots$ by splitting a set of the previous partitioning as it is illustrated in the following example,

$$\mathcal{B}^1 = \{\overbrace{\{1\}}^{\mathcal{A}^4}, \overbrace{\{2, 3, 4\}}^{\mathcal{A}^3}, \overbrace{\{5, 6, 7\}}^{\mathcal{A}^2}, \overbrace{\{8, 9, 10\}}^{\mathcal{A}^1}\} \tag{13}$$

$$\mathcal{B}^2 = \{\overbrace{\{1\}}^{\mathcal{A}^5}, \overbrace{\{2\}}^{\mathcal{A}^4}, \overbrace{\{3, 4\}}^{\mathcal{A}^3}, \overbrace{\{5, 6, 7\}}^{\mathcal{A}^2}, \overbrace{\{8, 9, 10\}}^{\mathcal{A}^1}\} \tag{14}$$

The binning procedure transforms the very large sample space with billions of elements into a smaller probability space where statistical testing can be applied.

4.5 Statistical Guarantees

In addition to the binning procedure, the proposed evaluation method provides statistical guarantees. This is achieved by setting an error threshold, e_{thresh}, as a function of the number of samples, m, the number of partitions, $|\mathcal{B}|$, and a probability significance, δ. Then, we describe a learning estimator for the total variation error, T, with e_{thresh} as only sample access to q is provided. The difference of the total variation errors for two generative models can be expressed with T and e_{thresh} providing statistical guarantees.

The learning estimator allows us to calculate the total empirical variation error when the ground truth distribution, p is provided with sample access to the learned distribution, q. There

are m samples from q ($\{\tilde{x}_i\}_{i=1}^m \tilde{x}_i \sim q$). There are m samples that approximate q. The estimator is defined as the average distance between the probability mass function and the sampled $\tilde{q}$.

$$d_{TV}(p, \tilde{q}) \triangleq T = \frac{1}{2} \sum_{x \in \Omega} |p_x - \tilde{q}_x|, \tag{15}$$

$$\text{where } \tilde{q}_x = \frac{1}{m} \sum_{i=1}^{m} \mathbb{1}[\tilde{x}_i = x] \tag{16}$$

From the literature [10], the learning estimator can then be related to the total variation error and the error threshold, e_{thresh}. Given a discrete distribution p with associated a partitioning of the sample space $\mathcal{B}$, m samples from a distribution q with the same partitioning of the sample space $\mathcal{B}$, and an error tolerance $\epsilon_{thresh} \in (0, 1]$, the random quantity T will fall within the following interval:

$$T \in [d_{TV}(p, q) - \epsilon_{thresh}, d_{TV}(p, q) + \epsilon_{thresh}] \tag{17}$$

with at least $1 - \delta$ probability provided that

$$\epsilon_{thresh} \geq \max(\sqrt{\frac{|\mathcal{B}|}{m}}, \sqrt{\frac{2\ln(2/\delta)}{m}}). \tag{18}$$

Thus if we are comparing two distributions, q_1 and q_2, we would be able to draw statistical guarantees as follows. Given a discrete distribution p with the associated partitioning of the sample space $\mathcal{B}$, and m samples from the distributions q and q_2 with the same partitioning of the sample space $\mathcal{B}$, denote by T_{q_1} and T_{q_2} the empirical total variation estimators of q_1 and q_2, respectively. For an error tolerance $\epsilon_{test} \in (0, 1]$ such that (18) holds for a selected constant $\delta \in (0, 1)$, the random quantity $T_{q_1} - T_{q_2}$ will fall within the following interval:

$$T_{q_1} - T_{q_2} \in [d_{tv}(p, q_1) - d_{tv}(p, q_2) \pm 2\epsilon_{thresh}] \tag{19}$$

with at least $(1 - \delta)^2$ probability. Now, when comparing the performance of generative models, the value for $T_{q_1} - T_{q_2} - 2\epsilon_{thresh}$ should be computed. If this value is greater than zero, we can conclude $d_{TV}(p, q_1) > d_{TV}(p, q_2)$ with statistical significance. Recall, d_{TV} is the empirical total variation error, thus the lower value indicates the better model.

4.6 Evaluation Procedure

The evaluation procedure primarily involves comparing the d_{TV} values for different granularity levels between generative models tasked to learn the same ground truth distribution p. Mathematically, this involves comparing $d_{TV}(p^{\mathcal{B}^i}, q_1^{\mathcal{B}^i})$ to $d_{TV}(p^{\mathcal{B}^i}, q_2^{\mathcal{B}^i})$ with the better performing model having the lower total variation error. However, there are some considerations to note when comparing. Firstly, recall $d_{TV}(p^{\mathcal{B}}, q^{\mathcal{B}}) \leq d_{TV}(p, q)$. We additionally recall equation 5 which states $d_{TV}(p^{\mathcal{B}^{i-1}}, q^{\mathcal{B}^{i-1}}) \leq d_{TV}(p^{\mathcal{B}^i}, q^{\mathcal{B}^i}) \cdots \leq d_{TV}(p, q)$. Together, these imply as the number of bins increases, we get a more accurate estimate of the total variation error d_{TV}. The number of bins, however, is constrained by the equation for e_{thresh} as we strive to keep e_{thresh} under 0.010. As an example, we consider figure 2 where this constraint would indicate the maximum number of bins should be around 6 or 7 for various m and δ values.

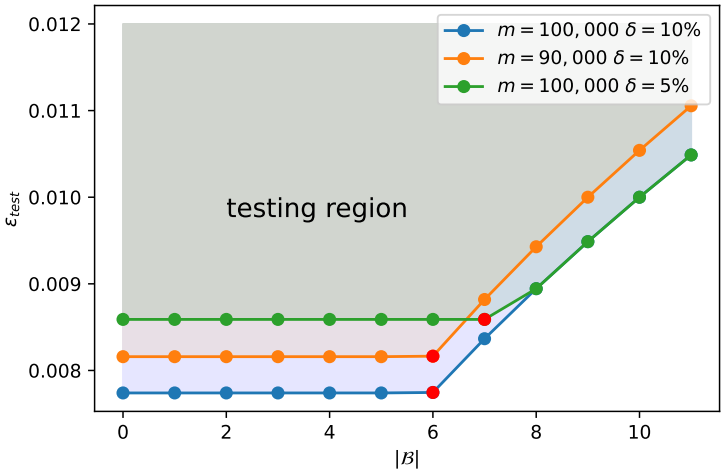


Figure 2: Error threshold, e_{thresh}, that we can obtain by varying the number of samples m and probability significance δ. The grey area denotes the possible error values.

4.7 Creation of the Synthetic Setting

In the synthetic setting, the ground truth distribution p can be chosen. Generative models are given samples with an underlying distribution p and are tasked with generating output samples from the learned distribution q. p must resemble a distribution a generative model would likely encounter. To do so, the sample space, $|\Omega|$ must be on the order of billions. Secondly, for the nature of categorical data, some elements x should be very likely to occur, some should be moderately likely, and others should have a zero probability of occurring. To resemble real world data, it is particularly important for $|\Omega^+|/|\Omega| \approx 0$, where $|\Omega^+|$ is defined as $\{p_x > 0\ \forall x \in \Omega\}$.

A representative task called SEQUENCE is created with these constraints. In SEQUENCE, elements x fall into one of three probability categories: *likely*, *rare*, or *zero*. SEQUENCE is crafted by considering combinations and permutations of length 6. There are $6^6 = 46,656$ combinations and $6! = 720$ permutations ($|\Omega| = 46656 + 720 = 47376$). All permutations are given a non-zero probability and are represented on the positive space, A_+. If the first element is larger than the last, permutations obtain a *likely* assignment and belong to A_{likely}. If the first element is smaller than the last, permutations obtain an *unlikely* assignment and belong to A_{rare}. Combinations are given a zero probability. Based on this arrangement, the probability mass function, p_x, can be defined as follows,

$$p(\mathbf{x}) = \begin{cases} \frac{3}{2*6!} & \text{if } \mathbf{x} \in A_{likely} = \{\mathbf{x}; \mathbf{x} \in A_+ \text{ and } \mathbf{x}_1 < \mathbf{x}_6\}, \\ \frac{1}{2*6!} & \text{if } \mathbf{x} \in A_{rare} = \{\mathbf{x}; \mathbf{x} \in A_+ \text{ and } \mathbf{x}_1 > \mathbf{x}_6\}, \\ 0 & \text{otherwise}. \end{cases} \tag{20}$$

The elements x in SEQUENCE are meant to resemble strands of amino acids in protein sequences. SEQUENCE satisfies the constraints of $|\Omega^+|/|\Omega| \approx 0$. Although this dataset is relatively small, it was used as part of the initial validation and testing process for the proposed evaluation metric.

4.8 Application to a Real World Setting

As mentioned, the objective is to apply the evaluation method for categorical generative models. One application of generative models where the data is categorical is in protein sequence modeling. The object is to generate 'valid' protein sequences given specific samples. In SEQUENCE, permutations and combinations were utilized to work towards a similar representative task. The real-world dataset Proteinnet7 is introduced as a larger venue for the comparison metric to be applied and is presently being used to train generative models for protein sequence modeling [8]. Primary structures of protein refer to sequences where the letters refer to 1 of 21 amino acids. Sequence length is fixed at 100. With this constraint, there are $|\Omega| = 21^{100} = 1.6 * 10^{132}$ possible primary structures possible. In Proteinnet7, there are $|\Omega_+|$ 12,575,738 primary structures for proteins. For the elements in p_x which have a non-zero probability mass, Ω_+, there are approximately 1.31% of samples that are 4x as likely, 5.28% of samples that are 3x as likely, 76.41% of samples that are 2x as likely and 17% of samples which are 1x as likely. The primary structures in Proteinnet7 with length 100 satisfy the constraints of $|\Omega^+|/|\Omega| \approx 0$ as $12 * 10^6/1.6 * 10^{132} \approx 0$. Further, $|\Omega|$ most certainly exceeds the constraint of 'in the billions' for its size. Experiments are currently underway with Proteinnet7 and will thus not be reported here.

4.9 Generative Models Used

Four state-of-the-art deep learning generative models are trained on the SEQUENCE task: GMCD, CNF, CDM, and argmaxAR. In GMCD [11], a diffusion-based model, random noise is added to the data and attempts to learn the reverse steps to generate the intended samples from the noise. The noise is based on Gaussian Mixtures. CNF, a normalizing flow, transforms the original distribution with multiple invertible and differentiable mappings into a more complex probability space. [12] CDM is also a diffusion-based model but is specifically for data of a categorical nature. [13] ArgmaxAR is also a normalizing flow where the argmax function is used as one of the functions to map the distribution to a more complex probability space. [14]

4.10 Metrics of Comparison

4.10.1 NLL

The negative log-likelihood is used as the first baseline comparison metric. The NLL is most commonly used and evaluates how likely the generated samples are to come from p, the ground truth distribution. The NLL is given by

$$NLL = \frac{1}{m} \sum_{i=1}^{m} \log q(x_i) \quad x_i \sim p \tag{21}$$

where m refers to the number of samples accessible. In the results, the exact NLL is reported when accessible.

4.10.2 Coverage Metrics

Coverage-based metrics intuitively measure how much the learned distribution, q, overlaps with the ground truth distribution, p, and vice versa. Precision and recall are two coverage metrics first introduced by [15]. To understand these metrics, we define S as the intersection of the support of p and the support of q, $S = \Omega^p \cap \Omega^q$. Precision is defined as the amount of q in p and is

given by $precision = \Omega^q/S$. Recall is defined as the amount of p in q and is given by $recall = \Omega^p/S$. Work proposed by [16] further improved these metrics developing alpha-precision, IP_α, and beta-recall, IR_β. IP_α and IR_β consider how close generated samples are to ground truth samples. They encompass a judgment of whether or not a new sample is truly generated by sampling from the learned distribution or simply a noisy version of an original.

Precision, Recall, IP_α and IR_β are metrics based in continuous domain and rely on a notion of distance between the samples generated. The categorical data used with the proposed evaluation method does not inherently have the desired continuous nature with a notion of distance. Thus, distance must be introduced into the SEQUENCE data. Data belonging to the same group, A_{likely}, A_{rare}, or A_{zero} is clustered together. A separation distance between the clusterings for A_{likely} and A_{rare} is induced by the $+1$ term in equation 22. A separation distance between the clusterings for (A_{rare} or A_{likely}) and A_{zero} is induced by the $+4$ term in equation 22. The hamiltonian distance is introduced such that we do not send the samples to the exact same location in space. The hamiltonian distance is based on the distance between two given samples and introduces the slight variation needed to form a cluster.

$$d(\mathbf{x}_i, \mathbf{x}_j) = \begin{cases} 0.01H(\mathbf{x}_i, \mathbf{x}_j) & \text{if } \mathbf{x}_i, \mathbf{x}_j \in A_{likely} \\ 0.01H(\mathbf{x}_i, \mathbf{x}_j) & \text{if } \mathbf{x}_i, \mathbf{x}_j \in A_{rare} \\ 0.01H(\mathbf{x}_i, \mathbf{x}_j) & \text{if } \mathbf{x}_i, \mathbf{x}_j \in A_{zero} \\ 1 + 0.01H(\mathbf{x}_i, \mathbf{x}_j) & \text{if } \mathbf{x}_{i/j} \in A_{likely} \text{ and } \mathbf{x}_{i/j} \in A_{rare} \\ 4 + 0.01H(\mathbf{x}_i, \mathbf{x}_j) & \text{if } \mathbf{x}_{i/j} \notin A_{zero} \text{ and } \mathbf{x}_{i/j} \in A_{zero} \end{cases} \tag{22}$$

.

4.10.3 Task-oriented metrics

The task-oriented metrics available such as the Inception Score (IS) [17] and Frechet Inception Distance (FID) [5] are not applicable for the categorical data available. Thus, no task-oriented metrics are used as baselines.

5 Results

5.1 Generated Samples

Four state-of-the-art generative models, argmaxAR, GMCD, CNF, and CDM are trained on the SEQUENCE task. Recall for the SEQUENCE task, $|A_+| = 720$, indicating there are 720 elements which has a non-zero probability mass. The ground truth distribution is pictured by the solid line. Notably, p is a stair distribution when in actuality it is a distribution with two levels: a top and bottom step. The vertical line down the middle is a drop meaning it does not exist in the true distribution. These generative models produce samples from the learned distribution pictured in figure 3. The CNF model does not perform well at following p. There are no 'steps' in the learned distribution. The other models, however, all follow p closely. Using the proposed evaluation metric and the baseline evaluation metrics, we shall be able to determine which model performs best.

5.2 Analysis with Proposed Metric

First, we examine the results using the proposed evaluation metric in figure 4. Here, $\delta = 10\%$, $|\Omega| = 6^6$, $m = 100,000$ samples, and $|\mathcal{B}| = 3$. The ground truth has the lowest total empirical

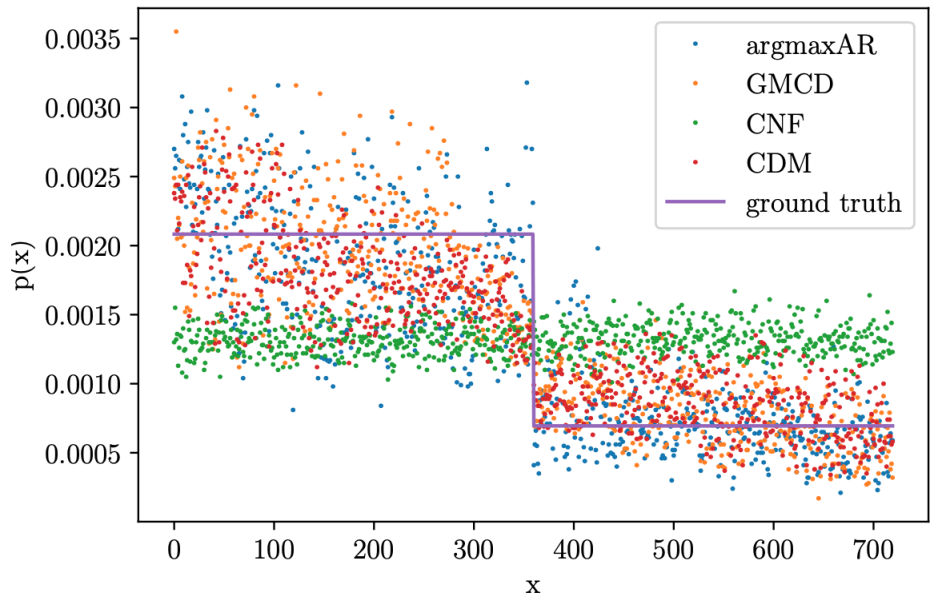


Figure 3: The empirical pmf q (m = 100,000, SEQUENCE task) is plotted against the samples generated from the four state of the art models.

variation error, $\hat{d_{TV}}(p^{\mathcal{B}^i}, q^{\mathcal{B}^i})$, over the binned distributions which is in accordance with what is anticipated. Additionally, we observe that the order of which model performs best is *consistent* as the number of bins, $|\mathcal{B}|$, increases. The model best model with the lowest $\hat{d_{TV}}(p^{\mathcal{B}^i}, q^{\mathcal{B}^i})$ is GMCD. We observe that as the number of bins, $|\mathcal{B}|$, increases, the error gets a bit higher. This can be attributed to the $\hat{d_{TV}}(p^{\mathcal{B}^i}, q^{\mathcal{B}^i})$ getting closer to the true error of $d_{TV}(p, q)$ as more bins, $|\mathcal{B}|$ achieves better representation of the non-binned distribution. Further, information is lost as we bin. Thus, it makes sense that the error would become less accurate with less bins. After GMCD, the ranking of generative models is argmaxAR, CDM, and CNF trailing significantly behind with its error. The poor performance of the CNF reported with $\hat{d_{TV}}(p^{\mathcal{B}^i}, q^{\mathcal{B}^i})$ matches the visual in figure 3. The proposed evaluation metric provides an interpretable ranking of generative models which remains consistent as the number of bins increases.

5.3 Analysis with Baseline Metrics

First, the performance of the evaluation metric is compared to the NLL in table 1. The first column specifies the total empirical variation error, $\hat{d_{TV}}(p, q)$, for the ground truth ranking. The empirical refers to the sample access we have to q rather than the distribution itself. Next, the NLL reports the correct ranking. No NLL is available on GMCD as it is not accessible highlighting one of the limitations of the NLL function. Then, $\hat{d_{TV}}(p^{\mathcal{B}^i}, q^{\mathcal{B}^i})$ is calculated and reported for $|\mathcal{B}| \in [3, 8]$ using the proposed evaluation metric. The proposed metric is able to retain the correct ranking across all binnings tested. The slope refers to the relative increase of the $\hat{d_{TV}}(p^{\mathcal{B}^i}, q^{\mathcal{B}^i})$ as $|\mathcal{B}|$ increases.

When comparing to the precison, recall, IP_α and IR_β metrics, it becomes difficult to decipher which model is outperforming the others based on figure 5. Notably, it is even difficult to state that CNF is performing poorly as compared to the NLL and total empirical variation error, $\hat{d_{TV}}(p^{\mathcal{B}^i}, q^{\mathcal{B}^i})$. The poor performance demonstrates these metrics are not applicable to the

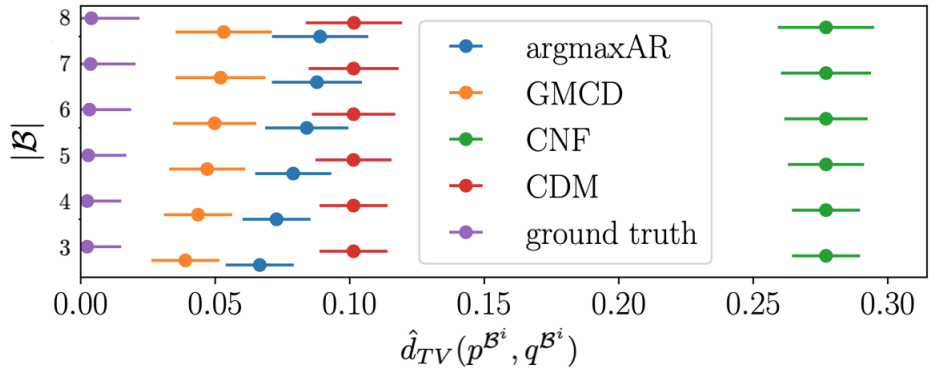


Figure 4: Probabilistic intervals from Theorem 3.1. with $\delta = 10\%$ probability. SEQUENCE exp., $|\Omega| = 6^6$, $m = 100,000$ samples, $S = 3$ regions.

Table 1: The SEQUENCE task is learned by the four generative models where $|\Omega| = 6^6$, $m = 100,000$ samples, $S = 3$ regions. A star (*) indicates statistical significance of 5%.

	$\hat{d}_{TV}$	NLL	B_3	B_4	B_5	B_6	$\mathbf{B}_7^*$	B_8	slope
GMCD	**0.149***	-	3.89*	4.36*	4.70*	4.98*	**5.20***	5.32*	0.284
argmaxAR	0.171*	**1.1104 ±0.1200**	6.66*	7.28*	7.90*	8.40*	8.78*	8.90*	0.463
CDM	0.183*	1.1111 ±0.1446	10.14*	10.14*	10.14*	10.14*	10.14*	10.16*	0.003
CNF	0.281*	1.1658* ±0.1167	27.70*	27.70*	27.70*	27.70*	27.70*	27.70*	0.000

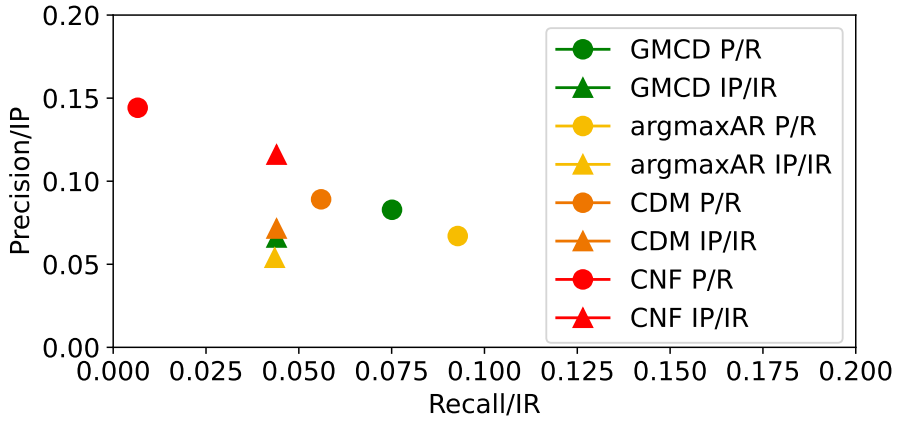


Figure 5: Precision/Recall, and IP_α/ IR_β metrics for the SEQUENCE task adapted to the continuous space.

categorical data setting where emphasis on different probability masses is more important than support coverage of the sample spaces. Although these metrics have advantageous applications, as demonstrated by [15] and [16], they remain at a disadvantage when evaluating the performance of generative models on categorical data.

6 Impact on Society and the Environment

With the potential to revolutionize many fields, generative models are becoming increasingly popular and trusted in society. For example, ChatGPT, a natural language processing model capable of generating human-like text, is transforming modern-day information gathering [3]. DALL-E, enabling the creation of images from textual description, is changing the nature of the marketing and fine art fields [18]. In the medical domain, Alphafold is a generative model which can predict the three-dimensional structure of proteins aiding drug discovery and development [19]. WaveNet is being used in the music industry to make music production more accessible by generating realistic speech and music samples [20]. Current evaluation metrics for these generative models, however, struggle to quantify the exact performance as existing metrics are task-specific, impossible to precisely compute, or poor at differentiating performance. This research seeks to explore the domain of evaluation methods for generative models to facilitate the development of better and safer generative models. Implicitly, the proposed evaluation method for categorical generative models has implications on the use of non-renewable energy resources, environment, societal safety, and societal impact.

Use of Non-Reusable Resources. There are few interactions with non-reusable resources in this research. The project does require compute resources, however, which, over time, can generate electronic waste. Thankfully, no new computational devices were purchased or discarded through the honours bachelor's thesis project minimizing the contribution to electronic waste.

Environmental Benefits. Generative models can be used to optimize energy consumption, reduce waste, and develop more sustainable practices in industries such as manufacturing, transportation, and agriculture. For example, deep generative models can perform energy consumption forecasting helping enterprises better anticipate demand surges for electricity [21]. Additionally, generative models can be used to create more accurate climate models helping to predict the impacts of climate change and aiding in the development of more effective policies for mitigating and adapting to its effects [22]. The use of generative models can help to minimize the impact of human activities on the environment and promote sustainable development. This work seeks to facilitate the development of better generative models, thus promoting solutions that most benefit the environment.

Safety. There are several safety concerns generative models raise in bias and fairness, privacy, and robustness which translate to their evaluation methods. Training generative models to avoid harmful biases is an active research field [23] [24] [25] [26]. Generative models further may output samples eerily close to the given input samples causing privacy concerns [27]. With this work, researchers can better evaluate how to introduce 'noise' into the generated samples without a significant negative impact on model performance. Lastly, evaluation methods that do not consider robustness can lead to unsafe generative models which produce incorrect or unreliable results.

Societal Benefits. Generative models have the potential to improve society in various fields, such as education, design, medicine, and climate science, by enabling more efficient, accurate, and creative solutions to complex problems. This work seeks to develop a tool to improve generative models enabling safer and more reliable models for society.

7 Conclusion

A proposed new evaluation method for categorical generative models has been presented and explored. The proposed algorithm is designed for very large sample spaces with synthetic experiments confirming its accuracy. The procedure is compared to the NLL and coverage-based metrics. We note the NLL is not tractable for the GCMD highlighting its limitations. For the other three models, the proposed metric achieves the same ranking as the NLL. The metric cannot easily be compared to the coverage-based metric as the categorical data does not evaluate well with coverage-based metrics additionally highlighting the limitation of the coverage-based metrics. The procedure provides statistical guarantees, is robust to different patterns of mismatch between p and q, and offers interpretable results. Future work will include running experiments on the much larger and real-life dataset, Proteinnet7, for the task of generative protein sequence modeling for drug discovery.

8 Statement of Learning

This past year in the Networks Research Lab, I have learned many new research and engineering skills. I began early on with the assignment to code backpropagation only to realize I, in fact, did not know machine learning nor python well enough to be able to do so. Now, I am not only confident in my coding abilities and understanding of machine learning but am more confident in my abilities to do statistics, read and summarize technical papers, generate plots and graphs to convey a specific message, use GitHub, understand complex problems, propose solutions, summarize work in a written and oral presentation format, provide weekly research updates, and conduct a literature review. I often commented to Florence that every time I returned to this project, I would learn something new about the proposed algorithm! My learning would not have been possible without the direct support of Florence Regol and Professor Coates. Below, I have summarized the general work and learning trajectories of each month of the honours electrical engineering thesis program.

- **In September,** I learned what a generative model was through meetings with Florence and was redirected to resources on YouTube and in blog posts to learn the statistical principles I would need to know to contribute to the project. I enrolled in ECSE 509 for the same reason. I met with Professor Coates to set goals for the year. Through the month, I learned how to most efficiently and effectively communicate with Florence given the time constraints of the semester. We additionally began running experiments in Python which also prompted me to learn how to use GitHub.

- **In October,** I continued running experiments learning how to properly debug with VSCode. I began working with matplotlib to plot graphs of interest in preparation for the ICASSP paper submission. This included exploring how to express bins and binned probability distributions. I additionally helped proof read the ICASSP paper and helped with writing the experimental section. I additionally made a graphic to summarize the proposed evaluation method.

- **In November,** following the ICASSP paper submission, we returned to the literatre for further algorithm development. I personally also returned to reading papers which were cited in the ICASSP paper to better understand the mathematical (statistical) underpinnings of the algorithm. Given I was now a few months into ECSE 509 and had

worked on the proposed evaluation method for some time, I was able to understand much more from the literature.

- **In December,** the literature review turned specifically into examining other evaluation metrics. I honed in on [15] and [16] by reading and rereading their papers and the papers the referenced until I understood the math to the best of my abilities. I worked closely with Florence during this time, asking many questions about the other metrics. Once an understanding of the other metrics was found, I turned to the code for these evaluation metrics.

- **In January,** I attempted to match the math from the papers with the code in the corresponding GitHub repositories. This was a bit convoluted and took me some time. With an understanding of the code, I was able to work with Florence to apply these metrics to the synthetic experiments we had applied the proposed evaluation metric on. I also used matplotlib to generate plot to compare the baseline evaluation metric against the proposed evaluation metric.

- **In February,** I began to take a look at algorithm scalability as $|\Omega|$ increases in size. I also presented the mid-semester project update to Professor Coates and to Professor Kanaan. I gave this presentation to the Networks Research Laboratory as well. I continued generating plots and tables and writing sections on the coverage-based metrics for submission to UAI. I also began investigating how real-world datasets could be used with the proposed evaluation metric.

- **In March,** I started experiments on Proteinnet7, a real-world dataset with primary structures of protein sequences on the order of billions in size. I ran into issues with my RAM and computation time with my personal computer hardware. I learned how to use GitHub better after trying and failing to upload more than 100 MB of data. I created the design day poster.

- **In April,** I presented the design day poster at McGill Design Day. I additionally wrote this thesis summarizing the work conducted over this project. I will submit a video to ICASSP for April 30th summarizing the paper for their virtual conference format.

- **In May,** I will prepare for the presentation of the research work at ICASSP in June and conclude any real-world data experiments with Proteinnet7.

- **Throughout the year,** I attended the weekly lab meetings of the Networks Research Laboratory. I found it insightful to observe how MS and PhD's convey their research updates. I learned a great deal of auxiliary information pertinent to the field of research through listening at these meetings. I additionally enjoyed the weekly presentations from members of the group on a technical topic. Each presentation was accompanied by a critique period which I have also found tremendously helpful to observe. In the lab, I have also been able to observe how a research lab operates in a longer-term setting. Lastly, I have been appreciative of the opportunity to observe what feedback people have received on their research and graduate school activities. This has been particularly insightful and I plan to carry those lessons onwards as I approach graduate school in the fall.

From the past year working on this project, a paper (Appendix A) has been accepted to IEEE ICASSP 2023, International Conference on Acoustic, Speech and Signal Processing. A secondary paper (Appendix B) has been submitted to UAI 2023, Uncertainty in Artificial Intelligence. This

work was presented in poster format at McGill Design Day 2023. I will submit a video summarizing the ICASSP paper by April 30. This June, I will present this work at ICASSP in Rhodes, Greece.

9 Appendix

Appendix A: Accepted to IEEE ICASSP 2023 - "Evaluation of Categorical Generative Models – Briding the Gap Between Real and Synthetic Data"
Appendix B: Submitted to UAI 2023 - "Evaluation of Categorical Generative Models: A Statistical Testing Framework for Very Large Sample Spaces"

References

[1] A. Strokach and P. M. Kim, "Deep generative modeling for protein design," *Current Opinion in Structural Biology*, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0959440X21001573

[2] Y. Asri, "Revolutionizing education: The power of generative ai in enhancing teaching," 2023. [Online]. Available: https://bootcamp.uxdesign.cc/revolutionizing-education-the-power-of-generative-ai-in-enhancing-teaching-a744a8e5265d

[3] OpenAI, "Chatgpt," 2023. [Online]. Available: https://openai.com/blog/chatgpt

[4] L. Theis, A. van den Oord, and M. Bethge, "A note on the evaluation of generative models," in *Proc. Int. Conf. Learning Representations ICLR*, 2016.

[5] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," in *Proc. Adv. Neural Info. Process. Syst. NeurIPS*, 2017.

[6] F. Regol, A. Kroon, and M. Coates, "Evaluation of categorical generative models – Bridging the gap between real and synthetic data," in *IEEE Intl. Conf. Acoustics, Speech, and Signal Proc. (ICASSP)*, 2023.

[7] S. O. Chan, I. Diakonikolas, P. Valiant, and G. Valiant, "Optimal algorithms for testing closeness of discrete distributions," in *Proc. Symp. on Discrete Algorithms (SODA)*, 2014.

[8] M. AlQuraishi, "Proteinnet: a standardized data set for machine learning of protein structure," *BMC Bioinformatics*, 2019.

[9] I. Diakonikolas, T. Gouleakis, D. M. Kane, J. Peebles, and E. Price, "Optimal testing of discrete distributions with high probability," in *Proc. ACM SIGACT Symp. on Theory of Comput.*, 2021.

[10] C. L. Canonne, "Topics and techniques in distribution testing: A biased but representative sample," *Foundations and Trends in Communications and Information Theory*, vol. 19, no. 6, pp. 1032–1198, 2022.

[11] F. Regol and M. Coates, "Diffusing gaussian mixtures for generating categorical data," in *Proc. AAAI Conf. on Artificial Intelligence*, 2023.

[12] P. Lippe and E. Gavves, "Categorical normalizing flows via continuous transformations," in *Proc. Int. Conf. Learning Representations ICLR*, 2021.

[13] E. Hoogeboom, D. Nielsen, P. Jaini, P. Forré, and M. Welling, "Argmax flows and multinomial diffusion: Learning categorical distributions," in *Proc. Adv. Neural Info. Process. Syst. NeurIPS*, 2021.

[14] E. Hoogeboom, D. Nielsen, P. Jaini, P. Forré, and M. Welling, "Argmax flows: Learning categorical distributions with normalizing flows," in *Proc. Symposium on Adv. in Appr. Bayesian Inference*, 2021.

[15] M. S. M. Sajjadi, O. Bachem, M. Lucic, O. Bousquet, and S. Gelly, "Assessing generative models via precision and recall," in *Proc. Adv. Neural Info. Process. Syst. NeurIPS*, 2018.

[16] A. Alaa, B. Van Breugel, E. S. Saveliev, and M. van der Schaar, "How faithful is your synthetic data? Sample-level metrics for evaluating and auditing generative models," in *Proc. Int. Conf. Machine Learning ICML*, 2022.

[17] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen, and X. Chen, "Improved techniques for training gans," in *Proc. Adv. Neural Info. Process. Syst. NeurIPS*, 2016.

[18] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with clip latents," *ArXiv*, vol. abs/2204.06125, 2022.

[19] A. W. Senior, R. Evans, and J. Jumper, "Improved protein structure prediction using potentials from deep learning," 2020.

[20] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," in *Arxiv*, 2016. [Online]. Available: https://arxiv.org/abs/1609.03499

[21] L. Zhu and Y. Huang, "Research on deep generative model application for shortterm load forecasting of enterprise electricity," *IOP Conference Series: Earth and Environmental Science*, p. 012113, 2021. [Online]. Available: https://dx.doi.org/10.1088/1755-1315/687/1/012113

[22] C. Besombes, O. Pannekoucke, C. Lapeyre, B. Sanderson, and O. Thual, "Producing realistic climate data with generative adversarial networks," *Nonlinear Processes in Geophysics*, vol. 28, no. 3, pp. 347–370, 2021. [Online]. Available: https://npg.copernicus.org/articles/28/347/2021/

[23] S. Zhao, H. Ren, A. Yuan, J. Song, N. Goodman, and S. Ermon, "Bias and generalization in deep generative models: An empirical study," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[24] M. Peychev, A. Ruoss, M. Balunović, M. Baader, and M. Vechev, "Latent space smoothing for individually fair representations," 2022.

[25] R. Srinivasan and K. Uchino, "Biases in generative art: A causal look from the lens of art history," in *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, 2021, pp. 41–51.

[26] E. M. Smith and A. Williams, "Hi, my name is martha: Using names to measure and mitigate bias in generative dialogue models," *arXiv preprint arXiv:2109.03300*, 2021.

[27] T. Cao, A. Bie, A. Vahdat, S. Fidler, and K. Kreis, "Don't generate me: Training differentially private generative models with sinkhorn divergence," *Advances in Neural Information Processing Systems*, vol. 34, pp. 12 480–12 492, 2021.