Signature Informed Sampling for Transcriptomic Data

Nikita Janakarajan
IBM Research Europe, ETH Zürich
Zürich, Switzerland
nja@zurich.ibm.com

Mara Graziani
IBM Research Europe
Zürich, Switzerland
Mara.Graziani@ibm.com

Maria Rodriguez Martinez*
IBM Research Europe
Zürich, Switzerland
maria.rodriguezmartinez@yale.edu

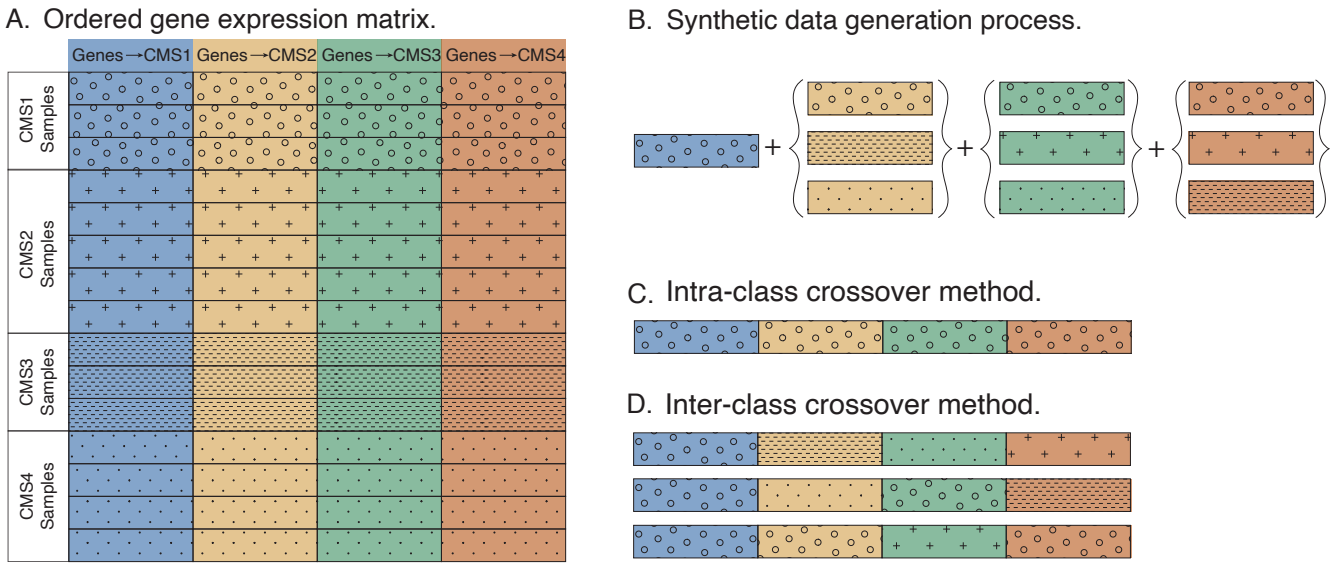

Figure 1: Crossover sampling strategy. A. Consider a gene expression matrix where each gene block, identified by colour, is the predictive signature of a phenotype, here CMS (colorectacl cancer subtype), identified by pattern. The rows represent patients with a given phenotype. B. To generate a new sample, we mix and match gene signature blocks under certain phenotypic constraints. C. Intra-class crossover sampling. A sample of class CMS1 is generated by sampling gene signature blocks from the subset of CMS1 patients only. D. Inter-class crossover sampling. A sample of class CMS1 is generated by sampling the CMS1 gene signature block from the subset of CMS1 patients only and each of the remaining gene blocks are sampled from patients who do not belong to the CMS subset that the gene block predicts.

Abstract

Working with transcriptomic data is challenging in deep learning applications due to its high dimensionality and low patient numbers. Deep learning models tend to overfit this data and do not generalize well on out-of-distribution samples and new cohorts. Data augmentation strategies help alleviate this problem by introducing synthetic data points and acting as regularisers. However, existing approaches are either computationally intensive or require parametric estimates. We introduce a new solution to an old problem - a simple, non-parametric, and novel data augmentation approach inspired by the phenomenon of chromosomal crossover. Based on the assumption that there exist non-overlapping gene signatures describing each phenotype of interest, we demonstrate how new synthetic data points can be generated by sampling gene signatures from different patients under certain phenotypic constraints. As a case study, we apply our crossover method to transcriptomic data of colorectal cancer. Through discriminative and generative experiments on two different datasets, we show that our method improves patient stratification by generating samples that mirror biological variability as well as the models' robustness to overfitting and distribution shift. Our approach requires little to no computation, and outperforms, or at the very least matches, the performance of established augmentation methods. The code is hosted on GitHub (click), and includes a link to the augmented datasets for reproducibility.

CCS Concepts

• **Applied computing** → **Computational transcriptomics**.

Keywords

Data Augmentation, Transcriptomics, Machine Learning

1 Introduction

The application of machine learning in biomedicine, particularly for classification tasks, has become increasingly popular over the last decade [39, 48]. High-throughput sequencing has provided access to transcriptomic data, catalyzing an increase in machine learning research aimed at pattern discovery, phenotype classification, disease subtyping, survival analysis, and more. However, the inherent challenges in this domain persist, notably the high

*Currently at Yale School of Medicine

dimensionality of data and limited sample numbers [15]. These challenges make machine learning models prone to overfitting, and reduce their ability to generalize to new datasets [23], even from different institutions [57].

To counter these undesired effects, we propose a signature-based crossover sampling approach that is computationally efficient and requires no distribution parameter inference. The crossover sampling method leverages phenotypic gene signatures – sets of genes with distinct expression patterns that hold prognostic, diagnostic or predictive value – to drive the augmentation. Gene signatures are invaluable for phenotypic predictions, and are especially useful in cancer subtyping [18, 50, 56, 62, 63]. They can be leveraged for dimensionality reduction, where each patient is represented by the set of all gene signatures associated with the phenotypes instead of the entire 60,000 genes. Although gene signatures reduce the dimension of the feature space, machine learning models still struggle with this data due to limited patient samples. Here, we present a solution to this problem by leveraging identified gene signatures to generate new data. Specifically, we make the following assumptions: (i) Highly predictive gene signatures are available for each variant of the phenotype of interest; (ii) Gene signatures of phenotype variants are non-overlapping; (iii) The genes associated with each gene signature can interact with other genes from the same signature (directly or indirectly).

Inspired by chromosomal crossover, our sampling method performs crossovers between patients' gene signatures within and between phenotype variants to generate new observations. We evaluate our method against standard techniques, including random oversampling, the current state-of-the-art SMOTE [10], and parametric approaches like negative binomial and Poisson sampling. To demonstrate the effectiveness of our sampling methods, we use the TCGA COADREAD dataset. We assess the generalisation performance on the out-of-domain CPTAC COAD dataset as it shows a shift in the distribution from TCGA. The method successfully addresses class imbalance, improves model performance, and offers a promising solution for data augmentation in the biomedical domain. We envision the method to be applicable in modelling mixed phenotype scenarios, particularly useful for cancer heterogeneity studies, and in developing models robust against class imbalance, a problem that plagues the biology domain. The crossover sampling method offers versatility for various scenarios, making it a promising addition to transcriptomic data augmentation methods.

Our contributions are as follows:

(1) We introduce a novel, non-parametric data augmentation method and benchmark it against conventional techniques;
(2) We demonstrate the utility of the negative binomial and Poisson distributions as data augmentation strategies, proposing a modification in the sampling strategy;
(3) We highlight the influence of augmentation size on model performance through a series of experiments in both discriminative and generative settings.

2 Related Work

Multiple techniques for dimensionality reduction and data augmentation have been developed to mitigate overfitting and enhance generalisation. On one hand, dimensionality reduction methods help by only selecting highly informative features [13, 51], or combinations of features such as in Principal Component Analysis (PCA) [29], Non-Negative Matrix Factorisation (NMF) [35], and Autoencoders [26]. On the other hand, data augmentation generates new data points from existing ones, addressing dataset imbalances and under-representation. Common data augmentation techniques involve noise injection-based methods, particularly in the imaging domain [11, 54]. The noise perturbation prevents complex models from memorizing the training data [4]. Applying noise injection to transcriptomic and biological data, however, requires caution since the noise may potentially disrupt the biological signal. For this, other augmentation approaches were proposed for transcriptomic data, including parametric and non-parametric strategies. Parametric methods rely on population statistics and data models. The Poisson distribution is a widely used distribution to model RNA-Seq count data [8, 40, 61], but assumes equal variance and mean. This limits its effectiveness for count data with a large support [43] and fails to capture over-dispersion in the data. The negative binomial distribution, also known as the gamma-Poisson distribution, is another widely used distribution that captures the over-dispersion of RNA-Seq count data [38, 49].

To solve class imbalance, augmenting transcripomic data using parametric methods requires class-specific parameter estimation. Challenges arise when the data distribution support is too large, leading to low data density in some regions and potentially generating uninformative data distributions. Conversely, limited support can result in insufficient sample diversity. Additionally, out-of-distribution data points can also impact parameter inference, rendering these methods sensitive to outliers and measurement errors [25]. To solve class imbalance, augmenting transcripomic data using parametric methods requires class-specific parameter estimation. Challenges arise when the data distribution support is too large, leading to low data density in some regions and potentially generating uninformative data distributions. Conversely, limited support can result in insufficient sample diversity. Additionally, out-of-distribution data points can also impact parameter inference, rendering these methods sensitive to outliers and measurement errors [25]. Non-parametric techniques, such as random oversampling [6, 17, 28], do not have this issue, as they bypass distribution assumptions. While effective for constructing accurate classifiers [3], these methods cannot generate new samples to increase sample diversity. To remedy this, [33] proposed a weighted mixing of observations. However, assigning unique labels to such samples can be challenging, making this approach less suitable for supervised learning.

Oversampling by interpolation is another popular method of augmenting biological data. Synthetic Minority Oversampling Technique (SMOTE) [10] and its variants [3, 22, 24, 34, 37, 60] are prime examples of this sampling strategy. This strategy has achieved state of the art performances on tasks affected by class imbalances. However, it has been found that for high-dimensional datasets SMOTE does not attenuate the bias towards the majority class [5]. Furthermore, when SMOTE is applied to high-dimensional data, it preserves the expected value of the minority class while decreasing its biological variability, which can be problematic for classifiers that rely on class-specific variances [5]. Moreover, SMOTE can

be computationally demanding due to extensive pairwise distance calculations.

Alternative methods for data expansion include aggregation and the generation of synthetic samples *in silico* [27, 52, 58]. Generative adversarial network (GAN) [19] and its variants are used to create synthetic gene expression samples [9, 33, 41, 59], but are computationally intensive, difficult to train, and require detailed knowledge about gene regulatory networks, in addition to requiring a large amount of data, which is difficult to obtain, particularly in cancer studies.

3 Proposed Augmentation Methods

3.1 Crossover sampling

Inspired by chromosomal crossover during meiosis, phenotype-driven sampling aims to create new data points from existing data points while increasing diversity in the generated samples. Simply put, to generate new samples for a given phenotype, we leverage the high informativeness of the associated gene signatures for that particular phenotype, while assuming that the signatures for the other phenotypes in that sample are less informative for the given phenotype. We investigate two ways of augmenting the data samples:

(1) *Intra-class crossover:* Crossing over between samples having the same phenotypic variant.
(2) *Inter-class crossover:* Crossing over between samples across all phenotypic variants.

In the following sections, we describe the method through a practical example on the gene expression of patients with colorectal cancer, using the Consensus Molecular Subtype (CMS) as phenotype. This phenotype has four variants, namely, CMS1, CMS2, CMS3, and CMS4 [21], and each CMS class has a set of 10 genes that are highly predictive of it [7]. This greatly reduces the dimensionality of our gene expression dataset and gives us a panel of 40 genes that can be used for classifying patients into the 4 colorectal cancer subtypes. We describe the two methods of crossing over using this dataset as our premise. Formal definitions of the methods, including their runtime complexities, can be found in Appendix A, B and Section 5.5, respectively.

3.1.1 Intra-class crossover sampling. To limit the likelihood of mixing phenotypes, and hence introducing samples of dubious labels, we first perform crossing over between samples belonging to the same phenotype. Thus, in this setting, new samples are generated only from samples having the same phenotype and so naturally, the synthetic sample inherits that phenotype. The crossing-over between samples is done at the gene signature level.

Figure 1C illustrates the process of intra-class crossover sampling. Consider an ordered gene matrix where the patients and the genes are grouped according to their CMS class. To generate a new CMS1 sample, we randomly sample each of the four gene signature blocks from the subset of all CMS1 patients only.

Since we sample the blocks only from the subset of patients belonging to CMS1, the expression values of genes associated with the other CMS types are typical of what is observed in a CMS1 patient. This process is repeated for the other CMS types, that is, a CMS2 patient is sampled only from the subset of patients belonging to CMS2, and so on.

3.1.2 Inter-class crossover sampling. Inter-class crossover sampling aims at utilising the entire dataset to create new samples, in contrast to using a phenotype specific subset in intra-class crossover sampling. In this method, we investigate the effect of sampling signature blocks across all samples as opposed to just a subset of samples. To avoid ambiguous labels, we add a constraint that prevents sampling the defining signature block from a sample unless it belongs to the phenotype to be sampled.

This method is illustrated in Figure 1D in the context of CMS classes. When creating a CMS1 sample, we first sample the defining signature block from the set of all CMS1 patients. This ensures the predictive signature block has typical values as seen in a CMS1 patient. The remaining gene blocks are sampled from the subset of all patients excluding those that belong to the class associated with the gene block being sampled. This prevents any ambiguity in class label assignment. For example, the CMS2 block can be sampled from all patients except those belonging to CMS2 since the expression values of the CMS2 block in the other patients are in a range that is not typically predictive of CMS2.

3.2 Modified parametric sampling methods

Parametric methods such as the gamma-Poisson and Poisson distributions are popular augmentation methods for RNA-Seq data. In our experiments, we modify these two methods to make them a stronger competitor against our crossover methods in solving the class-imbalance problem. Briefly, we modify the methods such that new distributional parameters are estimated from randomly sampled sub-populations for every new sample that is generated. A more detailed explanation is described in Appendix C and D.

3.2.1 Modified Gamma-Poisson Sampling. The negative-binomial (NB) distribution is a commonly used parametric method for augmenting RNA-Seq data. The negative-binomial (NB) distribution is typically expressed as a gamma-Poisson mixture [2, 20] to make it more tractable. In this formulation, the rate parameter λ of the Poisson distribution is a random variable sampled from a gamma distribution parameterised by the shape (α) and rate (β) parameters, which are estimated from the observed population. Our proposed modification to the sampling strategy is to create a mixture of gamma-poisson distributions. To generate every new observation, a subset S of samples is randomly chosen to estimate the α and β parameters from its mean μ and variance σ^2 as shown in Equation 1. These parameters initialise the gamma distribution. Since the augmentation is aimed at solving class imbalance, the subset is created from samples belonging to the same class. Thus, the newly generated observation will have the same label as the subset. The size of the subset $|S|$ is a hyperparameter defined by the user. The random variable sampled from this gamma distribution is used to initialise the Poisson distribution from which a new observation is sampled. This process is repeated n times to generate n new observations.

$$\mu = \frac{1}{|S|}\sum_{x\in S} x, \quad \sigma^2 = \frac{1}{|S|}\sum_{x\in S}(x-\mu)^2$$
$$\alpha = \frac{\mu^2}{\sigma^2}, \quad \beta = \frac{\mu}{\sigma^2} \tag{1}$$

3.2.2 Modified Poisson Sampling. Similarly to the gamma-Poisson strategy, a subset is first sampled from the class to be augmented, after which the mean is computed and set as the rate parameter λ for the Poisson distribution as shown in Equation 2.

$$\lambda = \frac{1}{|S|}\sum_{x\in S} x\ , \quad Y_j \sim Poisson(\lambda_j) \tag{2}$$

4 Experimental Setting

4.1 Datasets

TCGA COADREAD (Colon Adenocarcinoma and Rectal Adenocarcinoma) [45] and CPTAC [16] are used for all experiments. TCGA COADREAD is designated for training and in-domain testing, while the CPTAC Colon Cancer gene expression data are exclusively used for out-of-domain validation. As an initial preprocessing step, we filter out replicates and only consider patients with primary tumours and available CMS labels. Summary of the class counts of the datasets is shown in Appendix Table 2. Additionally, since we are interested in colorectal cancer subtyping, we only consider the 40 signature genes of the subtypes [7]. Details on dataset augmentation process are described in 4.2.

4.2 Data Augmentation Process

We compare against four baseline data augmentation strategies - (i) Modified Gamma-Poisson sampling (Mod. GP), (ii) Modified Poisson sampling (Mod. Poisson), (iii) Synthetic Minority Oversampling Technique (SMOTE) [36], and (iv) Random oversampling (Replacement). The train and test splits are created from the TCGA COADREAD gene expression count data. Each training data split is augmented by the aforementioned methods. Note, after the augmentation we perform FPKM normalisation (Fragment Per Kilobase per Million mapped reads) and log transformation as per [12], based on the training split. We consider 3 class sizes for augmentation - Max, 500 and 5000. In the 'Max' setting, only the minority classes are oversampled to match the size of the majority class. For class sizes 500 and 5000, all classes are augmented until the number of samples in each class reaches 500 and 5000, respectively. In addition to testing the generalisability of the augmentation methods, the different sizes also provide insights into the effect of size on the quality of the augmented data. We also include the unaugmented log-FPKM normalised gene expression data from TCGA COADREAD as a baseline. This dataset is referred to as the "Unaugmented" dataset henceforth. A visualisation of the augmented data is provided in Appendix E. Details on hyperparameter selection for the modified gamma-poisson and modified poisson are described in Appendix F.

4.3 Models

For all discriminative tasks, we consider 5 different classifiers, namely, Logistic Regression (LR), K-Nearest Neighbours (KNN), Support Vector Machines with RBF kernel (SVM-RBF), Explainable Boosting Machines (EBM), and Random Forests (RF). For the generative tasks, we use a variational autoencoder [32]. Implementation details for the discriminative and generative experiments are in Appendix G.

4.4 Evaluation Criteria

We report the balanced accuracy (Equation 7) for all classification tasks on the designated TCGA test set and CPTAC data to ensure directly comparable performance across methods. If the support samples for each class exhibit sufficient and meaningful diversity, the test classification performance on the augmented data is expected to outperform the one on the unaugmented dataset. We perform a 5-fold stratified cross-validation repeated 5 times for each augmentation method and augmentation size.

5 Results

We use colorectal cancer subtypes (CMS) as a showcase example, specifically on the datasets described in subsection 4.1 and subsection 4.2. The experiments are centered around the downstream task of CMS subtype prediction and its associated clinical variables. These tasks demonstrate the utility of our augmentation methods in diverse settings. Implementation details of all experiments are described in Appendix G.

5.1 Classification Performance

We compare the different augmentation methods in predicting the CMS class across the three class sizes - max, 500 and 5000. We report the average balanced accuracy across all classifiers mentioned in subsection 4.3 as the aggregate measure on the in-domain TCGA test set (Figure 2) and the unseen, out-of-domain CPTAC dataset (Figure 3). Tables for performances of individual classification models with each augmentation method and class size is shown in Appendix Table 5 and Table 6.

From Figure 2, the modified gamma-poisson augmentation method is best suited for in-domain data. At class size Max, the average balanced accuracy is 87.57% ± 0.029, at class size 500 it increases to 88.18% ± 0.028, and it begins to saturate at class size 5000 with 88.43% ± 0.029.As expected, the choice of class size impacts the effectiveness of the data augmentation methods. Interestingly, the distribution-based modified gamma-poisson perfoms the best when the number of augmented samples is pushed to the extreme, benefiting from the increased class size in contrast to other methods. However, the trend indicates that over-augmentation offers little to no benefits at the expense of increased computation ($\approx$ 0.3% performance gain from class size 500 to 5000).

On the out-of-domain dataset, the inter-class crossover method generalises the best among all augmentation methods as seen in Figure 3. In particular, the inter-class crossover reports the highest performance overall for the class sizes 500 (65.34% ± 0.032) and 5000 (66.45% ± 0.019), resulting in a 5% − 6% increased accuracy over unaugmented data (average balanced accuracy 59.9% ± 0.0297) and 5% − 8% increased accuracy over data augmented by SMOTE (59.28% ± 0.0328 and 58.51% ± 0.0349 respectively) and random oversampling (58.93% ± 0.0313 and 59.18% ± 0.0358 respectively) for the two class sizes respectively.

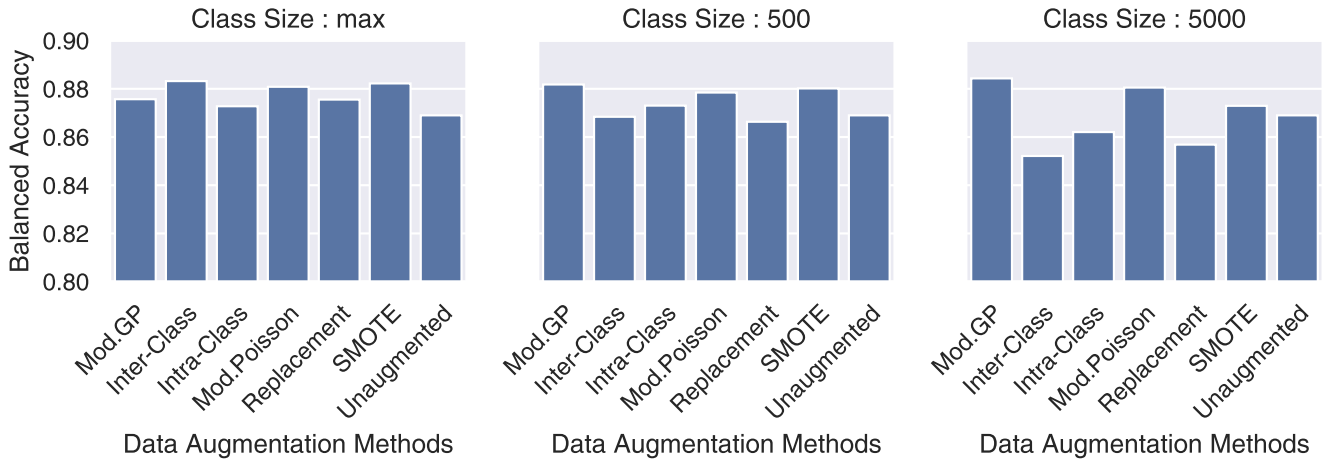

Figure 2: Averaged balanced accuracy on in-domain TCGA test set across classification models for every augmentation method and class size. The trend indicates that over-augmentation relative to dataset size offers little to no benefits and distribution based methods like modified gamma-poisson are overall a good candidate for tasks associated with in-domain data generation.

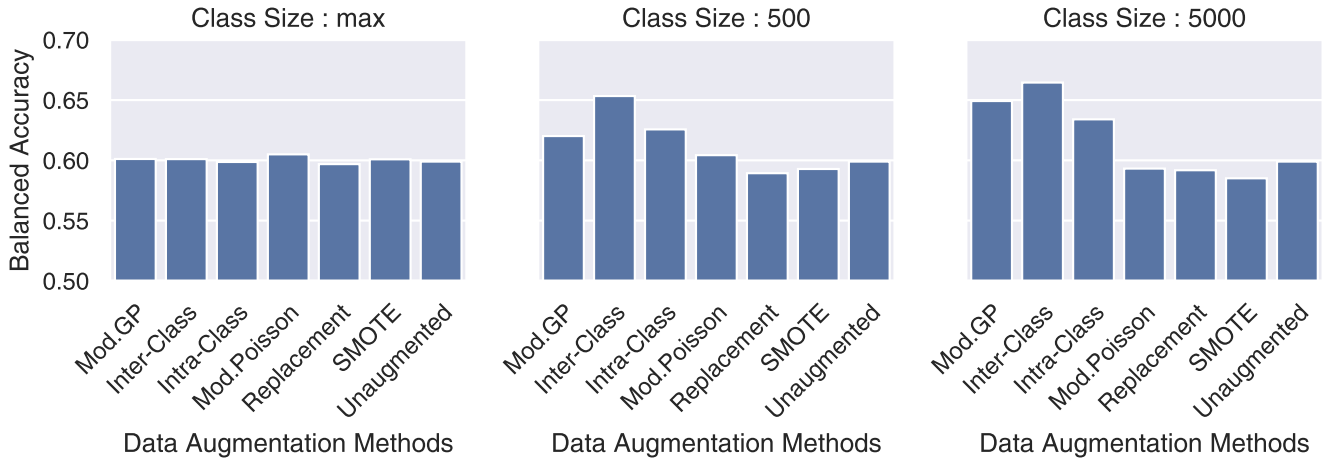

Figure 3: Averaged balanced accuracy on out-of-domain CPTAC dataset across classification models for every augmentation method and class size. The trend indicates that inter-class crossover method achieves the best generalisation performance and is a good candidate for tasks needing generalisation as it can achieve the best accuracy with fewer augmented samples compared to modified gamma-poisson.

5.2 Effect of Real Data Size

To evaluate the impact of initial dataset size to the different augmentation strategies, we select diverse sub-portions of the datasets, based on an increasing percentage of available real training data - namely 10, 20, 30, 40 and 50 percent. The real data at different percentages were sampled in a stratified manner to reflect the class imbalances present in the original datasets. This simulates the "very low data regime" scenario where the available data to train models is increasingly reduced. We repeated the classification experiment at each percentage and tested the models on the in-domain TCGA test set and unseen out-of-domain CPTAC dataset at different augmented class sizes. Detailed results of individual classification models with each augmentation method and class size for the 10% real data scenario is shown in Appendix Table 7 and Table 8.

Figure 4 shows the in-domain averaged balanced accuracy across all classification models considered for the three class sizes at different percentages of real data in training. The trends in all three subplots are consistent with the fact that as we increase the number of real data observations, the performance of the classification models improve. We see again that over-augmenting the data only has a limited improvement for the distribution based methods. For instance, for the modified gamma-poisson method on 10% of the real training data, the average balanced accuracy associated increases

from 82.444% ± 0.0528 to 82.54% ± 0.0571 when augmenting from class size 500 to 5000.

Figure 5 shows the averaged balanced accuracy across all classification models on the out-of-domain CPTAC dataset. At class sizes 500 and 5000, the inter-class crossover method achieves the best performance. Even when only 10% of the real training data is used, the performance of the inter-class crossover is as high as that of using 100% of the data. The second best method is modified gamma-poisson, which achieves a similarly high performance for class size 5000 (65.46% ± 0.0515). However, the same can already be achieved by our inter-class crossover sampling with class size 500 (65.55% ± 0.0492), making it the best method for generalisation. Also on CPTAC, when 100% real training data is used for augmentation, the performance increases with class size for the cross-over methods, but the improvement is limited (< 1%).

In conclusion, over-augmentation does not significantly help performance other than adding to computational runtime, and, for some augmentation methods, it can even be detrimental to the performance. We find that augmenting each class to size 500 proves to be optimal for all augmentation methods. Henceforth, we carry out further experiments using this class size.

5.3 Mixing Gene Blocks is not Random Noise

We consider the scenario where only 10% of the real training data (≈ 40 samples) is available. Since we retain the gene blocks as is, we do not destroy phenotype-specific information through mixing. We demonstrate this point by showing that key clinical variables of colorectal cancer subtypes, namely, microsatellite instability (MSI) and CpG Island Methylator Phenotype (CIMP), are still predictable. Predicting these variables, among others such as demographic factors, genomic markers and treatment response, is relevant for patient prognostics, disease monitoring and disease subtyping [14, 30, 44, 47, 53, 55]. Since we cannot infer MSI and CIMP labels for augmented data points, we first train a variational autoencoder using the augmented data (40-dimensional), retrieve the compressed embeddings (4-dimensional) of real data points and train classifiers on these embeddings. Thus, we indirectly evaluate the augmentation methods' ability to preserve information from the subtypes other than CMS typing.

Aggregated results are shown in Table 1, where we report the average balanced accuracy across all classifiers in predicting the MSI and CIMP status of patients. The inter-class and intra-class augmentation methods show a significant improvement over unaugmented data - 12% in-domain and 20% out-of-domain for MSI prediction, and 18% in-domain for CIMP prediction. These results prove that our crossover-based augmentation methods do not result in random samples whose only purpose is to introduce noise in the data for adversarial training. The mixing of gene blocks between samples does not drastically affect the predictiveness of other variables associated with these colorectal cancer subtypes. Results for each classifier and augmentation method can be found in Appendix Table 11, Table 12 and Table 13.

5.4 Generation quality

Lastly, we evaluate the augmentation methods in the generative modelling task with only 10% of given real training data. Here, we evaluate how effectively the augmention contributes to estimating a parametric representation of the data. Our hypothesis is that greater diversity in the input samples will lead to more accurate parametric estimates of the underlying distribution. To verify this hypothesis, we extract test data embeddings from a pre-trained variational autoencoder, setting class size to 500 and using 10% of the real training data. We compute the mean and standard deviation of these embeddings grouped by their true CMS class and intialiase a Gaussian-Mixture model. From this model, we generate new datapoints and their associated labels. This procedure is done for both TCGA and CPTAC test data. These newly generated data are then passed through our previously trained classifiers from Section 5.1. We expect a high classification accuracy to indicate that the VAE is able to learn the underlying distribution. This is possible only if the data on which it was trained is of sufficient size and quality.

Figure Figure 6 shows the average balanced accuracy achieved by each model on generated samples modelled on in-domain TCGA test set. As expected, augmentation is absolutely necessary in this low-data regime to learn anything useful from the data. On average, almost all methods are on-par with each other, with an average difference of around 2% between the methods. The modified gamma-poisson augmentation method achieves the best performances with the classification models, with an average accuracy of 77.71% ± 0.0536.

However, a considerable performance difference is seen on the out-of-domain CPTAC dataset as shown in Figure Figure 7. The inter-class crossover method (average balanced accuracy 66.36% ± 0.0631) performs the best over other methods, with an increase in

Table 1: Balanced accuracy scores of a 5x5 cross-validation for MSI and CIMP Classification on class size 500 with only 10% real training data. The best scores are highlighted in bold. For both MSI and CIMP classification, our crossover methods are consistently in the top 2 in both in-domain (MSI and CIMP) and out-of-domain validation (MSI only). The results show that our crossover methods significantl improve prediction power over unaugmented data.

Clinical Variable	Sampling Methods	Average Balanced Accuracy	
		TCGA	CPTAC
MSI	Mod. Gamma-Poisson	0.5912 ± 0.0576	0.6348 ± 0.0990
	Inter-Class (Ours)	0.6046 ± 0.0562	0.7039 ± 0.1198
	Intra-Class (Ours)	0.6054 ± 0.0594	**0.7086 ± 0.1087**
	Mod. Poisson	0.5860 ± 0.0584	0.6603 ± 0.0907
	Replacement	**0.6127 ± 0.0533**	0.6884 ± 0.1012
	SMOTE	0.5981 ± 0.0580	0.6490 ± 0.0972
	Unaugmented	0.4868 ± 0.0346	0.5023 ± 0.0350
CIMP	Mod. Gamma-Poisson	0.4962 ± 0.0571	–
	Inter-Class (Ours)	**0.5228 ± 0.0624**	–
	Intra-Class (Ours)	0.5102 ± 0.0628	–
	Mod. Poisson	0.4974 ± 0.0704	–
	Replacement	0.5095 ± 0.0654	–
	SMOTE	0.5066 ± 0.0711	–
	Unaugmented	0.3408 ± 0.0427	–

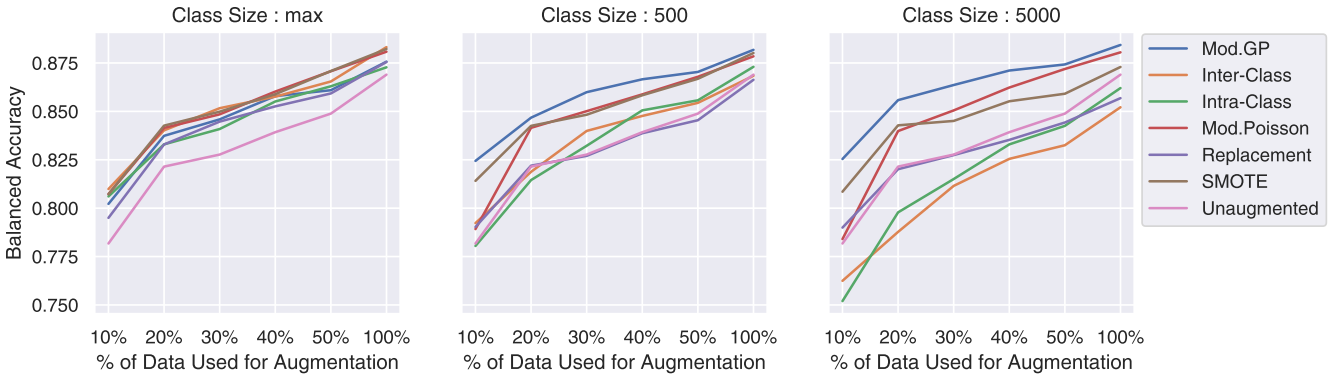

Figure 4: Trends in average classification performance on in-domain TCGA test set as the percentage of available real training data increases. Modified gamma-poisson sampling is resistant to over-augmentation.

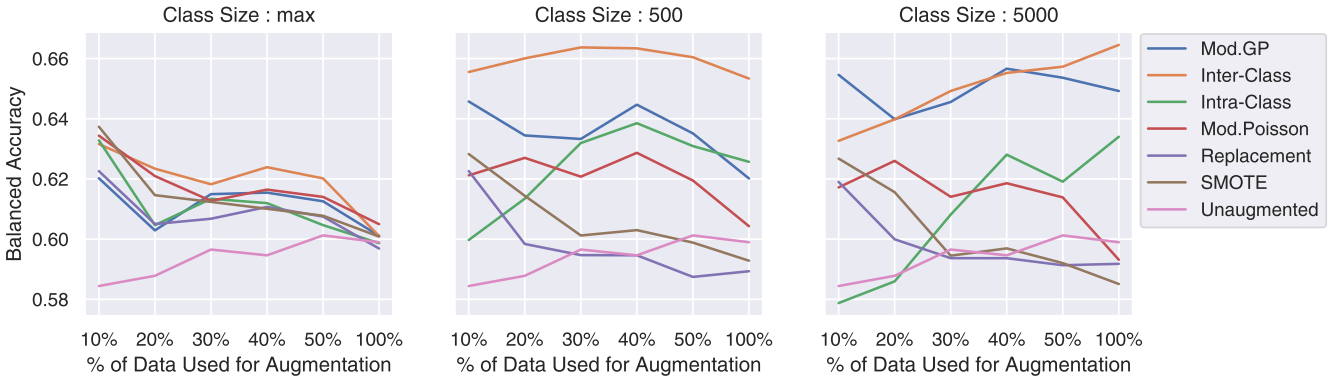

Figure 5: Trends in average classification performance on out-of-domain CPTAC dataset as the percentage of available real training data increases. Inter-class crossover sampling is resistant to over-augmentation and produces the best performance with fewer augmented data points compared to other methods.

accuracy of ≈ 35% over unaugmented data, ≈ 6% over modified gamma-poisson, modified poisson and replacement methods, and ≈ 5% over SMOTE. The results of the experiment are shown in Appendix Table 9 and Table 10.

5.5 Computational Complexity

Runtime complexity determines a method's efficiency and scalability, properties crucial for deployment in the real-world. The runtime complexity of the different methods is computed for generating samples of a given phenotype. The crossover methods both use `numpy.random` methods, namely `randint` for intra-class and `choice` for inter-class, to randomly sample signature blocks where the probability of sampling a block is uniformly distributed. Given that the time complexity depends on the number of samples generated, both methods have a runtime complexity of $O(pk)$ where p is the number of phenotypes (in the case of CMS, 4) and k is the number of new samples to be generated. Thus the complexity is linear in both k (if p is fixed) and p (if k is fixed).

The random oversampling method has a runtime complexity of $O(kn)$ since we generate k new samples and for each sample we go through the n values in the dataset once for sampling. Given $s = |S|$, the number of genes in the signature, the parametric methods have a runtime complexity of $O(psnk) + O(k)$, where p is the number of phenotypes, and n is the number of samples from which the distribution parameters such as mean and standard deviation are computed. The first term is the runtime complexity of computing the distribution parameters, where it is computed for each column in the matrix, and the second term is the runtime complexity of sampling values from the distribution. Given that s, p, n and k are all independent from each other, all three methods have a linear runtime complexity. SMOTE has a quadratic runtime complexity since it requires the computation of a pairwise distance matrix, and is dependent on the number of samples, n, in the dataset ($O(n^2)$).

6 Discussion

We introduced a novel, computationally efficient method for augmenting transcriptomic data that is inspired by chromosomal crossover. When distinct gene signatures represent target phenotypes, our approach allows us to generate synthetic data by sampling values across patients while adhering to specific phenotypic constraints.

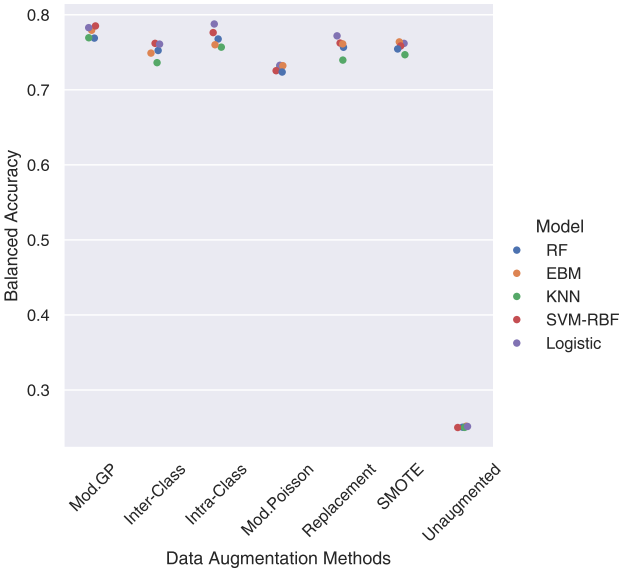

Figure 6: Classification performance of different models on newly generated samples modelled on in-domain TCGA test set. The VAEs generating the samples were trained on data augmented by the different methods as shown in the figure. Although the methods show a similar performance, on average, the modified gamma-poisson augmentation method is associated with the best score($77.71\% \pm 0.0536$**).**

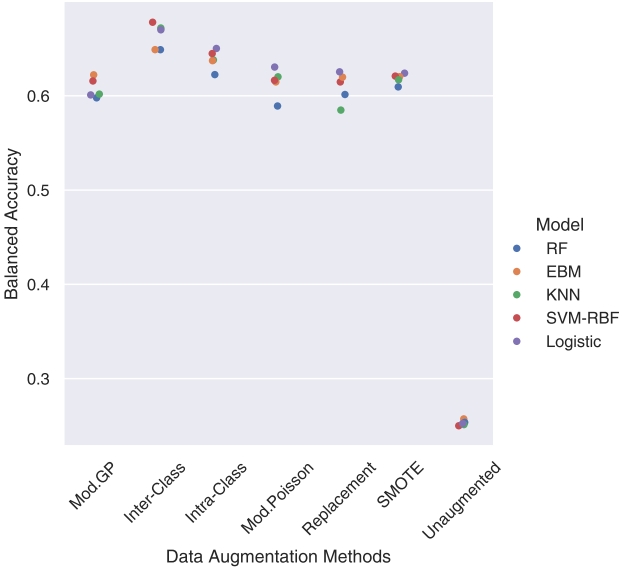

Figure 7: Classification performance of different models on newly generated samples modelled on out-of-domain CPTAC test set. The VAEs generating the samples were trained on data augmented by the different methods as shown in the figure. On average, the inter-class crossover sampling is associated with the best performance ($66.36\% \pm 0.0631$**).**

Results show that our augmentaion methods lead to the best model performance, irrespective of the selected models and tasks. Particularly, our methods seem to improve model robustness to overfitting and distribution shift. Such effectiveness is demonstrated by the higher performances on the classification experiments on both TCGA COADREAD and CPTAC COAD. Particularly, inter-class crossover consistently proves to be the best choice to achieve generalisation. Moreover, the modified gamma-poisson proves to be useful for downstream tasks needing in-domain data generation. Overall, our methods are efficient (i.e. linear cost) and lead to at least on-par performance, if not higher, than popular methods like SMOTE, which is computationally more expensive being quadratic in the number of samples.

A potential limitation of the intra- and inter-class crossover approaches is the assumption that gene signatures are known for a specific phenotype. However, there is increasing research on discovering gene signatures and one can assume that this knowledge will be inevitably be available in the future. Nevertheless, our method does not require perfect certainty or knowledge on the gene signatures as candidate signatures can be inferred through a prior analysis using univariate regression or differential expression analysis of genes. Another limitation is the reliance on non-overlapping gene signatures predictive of each phenotype. This can be overcome by treating dynamically the gene signature blocks instead of keeping them fixed in size and content. For example, the block could be defined by the patient itself.

We envision extending our approach to gene signatures with correlated genes, as classifiers can disentangle existing correlations. Although augmenting datasets with correlated classes may introduce challenges, we intend to explore this further in future work. Additionally, we plan to investigate the impact of our method when applied to the entire genome without the phenotype conditioning. This would remove the need for prior knowledge on the signature, but may present challenges due to the risk of spurious correlations and signal corruption. Despite these limitations, our sampling methods are an intuitive and simple application, are very efficient and brings on-par if not better performance than state-of-the-art methods, holding promise for heterogeneity studies and disease analysis. Data augmentation becomes almost instantaneous by cross-mixing samples, creating a landscape of mixed phenotypes, the diversity of which depends on the dataset size. This opens new opportunities in unsupervised and semi-supervised learning in scarce data regimes given fully unlabeled data.

References

[1] Abien Fred Agarap. 2018. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375* (2018).

[2] Tim Barry. 2020. Tim Barry: Gamma, Poisson, and negative binomial distributions. https://timothy-barry.github.io/posts/2020-06-16-gamma-poisson-nb/

[3] Gustavo EAPA Batista, Ronaldo C Prati, and Maria Carolina Monard. 2004. A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD explorations newsletter* 6, 1 (2004), 20–29.

[4] Christopher M Bishop et al. 1995. *Neural networks for pattern recognition.* Oxford university press.

[5] Rok Blagus and Lara Lusa. 2013. SMOTE for high-dimensional class-imbalanced data. *BMC bioinformatics* 14, 1 (2013), 1–16.

[6] Paula Branco, Luís Torgo, and Rita P Ribeiro. 2016. A survey of predictive modeling on imbalanced domains. *ACM computing surveys (CSUR)* 49, 2 (2016), 1–50.

[7] Steven A Buechler, Melissa T Stephens, Amanda B Hummon, Katelyn Ludwig, Emily Cannon, Tonia C Carter, Jeffrey Resnick, Yesim Gökmen-Polar, and Sunil S

Badve. 2020. ColoType: a forty gene signature for consensus molecular subtyping of colorectal cancer tumors using whole-genome assay or targeted RNA-sequencing. *Scientific reports* 10, 1 (2020), 1–13.

[8] James H Bullard, Elizabeth Purdom, Kasper D Hansen, and Sandrine Dudoit. 2010. Evaluation of statistical methods for normalization and differential expression in mRNA-Seq experiments. *BMC bioinformatics* 11, 1 (2010), 1–13.

[9] Poonam Chaudhari, Himanshu Agrawal, and Ketan Kotecha. 2020. Data augmentation using MG-GAN for improved cancer classification on gene expression data. *Soft Computing* 24, 15 (2020), 11381–11391.

[10] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. 2002. SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research* 16 (2002), 321–357.

[11] Phillip Chlap, Hang Min, Nym Vandenberg, Jason Dowling, Lois Holloway, and Annette Haworth. 2021. A review of medical image data augmentation techniques for deep learning applications. *Journal of Medical Imaging and Radiation Oncology* 65, 5 (2021), 545–563.

[12] Ana Conesa, Pedro Madrigal, Sonia Tarazona, David Gomez-Cabrero, Alejandra Cervera, Andrew McPherson, Michał Wojciech Szcześniak, Daniel J Gaffney, Laura L Elo, Xuegong Zhang, et al. 2016. A survey of best practices for RNA-seq data analysis. *Genome biology* 17, 1 (2016), 1–19.

[13] Manoranjan Dash and Huan Liu. 1997. Feature selection for classification. *Intelligent data analysis* 1, 1-4 (1997), 131–156.

[14] Zhenli Diao, Yanxi Han, Yuqing Chen, Rui Zhang, and Jinming Li. 2021. The clinical utility of microsatellite instability in colorectal cancer. *Critical reviews in oncology/hematology* 157 (2021), 103171.

[15] Sandrine Dudoit, Jane Fridlyand, and Terence P Speed. 2002. Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American statistical association* 97, 457 (2002), 77–87.

[16] Nathan J Edwards, Mauricio Oberti, Ratna R Thangudu, Shuang Cai, Peter B McGarvey, Shine Jacob, Subha Madhavan, and Karen A Ketchum. 2015. The CPTAC data portal: a resource for cancer proteomics research. *Journal of proteome research* 14, 6 (2015), 2707–2713.

[17] Bradley Efron and Robert J Tibshirani. 1994. *An introduction to the bootstrap*. CRC press.

[18] Todd R Golub, Donna K Slonim, Pablo Tamayo, Christine Huard, Michelle Gaasenbeek, Jill P Mesirov, Hilary Coller, Mignon L Loh, James R Downing, Mark A Caligiuri, et al. 1999. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *science* 286, 5439 (1999), 531–537.

[19] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial networks. *Commun. ACM* 63, 11 (2020), 139–144.

[20] Major Greenwood and G Udny Yule. 1920. An inquiry into the nature of frequency distributions representative of multiple happenings with particular reference to the occurrence of multiple attacks of disease or of repeated accidents. *Journal of the Royal statistical society* 83, 2 (1920), 255–279.

[21] Justin Guinney, Rodrigo Dienstmann, Xin Wang, Aurélien De Reynies, Andreas Schlicker, Charlotte Soneson, Laetitia Marisa, Paul Roepman, Gift Nyamundanda, Paolo Angelino, et al. 2015. The consensus molecular subtypes of colorectal cancer. *Nature medicine* 21, 11 (2015), 1350–1356.

[22] Hui Han, Wen-Yuan Wang, and Bing-Huan Mao. 2005. Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning. In *International conference on intelligent computing*. Springer, 878–887.

[23] Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. 2009. *The elements of statistical learning: data mining, inference, and prediction*. Vol. 2. Springer.

[24] H He, Y Bai, EA Garcia, and S ADASYN Li. 2008. adaptive synthetic sampling approach for imbalanced learning. IEEE international joint conference on neural networks. In *2008 (IEEE World Congress On Computational Intelligence)*.

[25] Dan Hendrycks and Kevin Gimpel. 2016. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136* (2016).

[26] Geoffrey E Hinton and Ruslan R Salakhutdinov. 2006. Reducing the dimensionality of data with neural networks. *science* 313, 5786 (2006), 504–507.

[27] Hai-Hui Huang, Hao Rao, Rui Miao, and Yong Liang. 2022. A novel meta-analysis based on data augmentation and elastic data shared lasso regularization for gene expression. *BMC bioinformatics* 23, 10 (2022), 1–24.

[28] Nathalie Japkowicz and Shaju Stephen. 2002. The class imbalance problem: A systematic study. *Intelligent data analysis* 6, 5 (2002), 429–449.

[29] Ian T Jolliffe. 2002. *Principal component analysis for special types of data*. Springer.

[30] Sanghee Kang, Younghyun Na, Sung Yup Joung, Sun Il Lee, Sang Cheul Oh, and Byung Wook Min. 2018. The significance of microsatellite instability in colorectal cancer after controlling for clinicopathological factors. *Medicine* 97, 9 (2018).

[31] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[32] Diederik P Kingma and Max Welling. 2014. Auto-Encoding Variational Bayes. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

[33] Magdalena Kircher, Elisa Chludzinski, Jessica Krepel, Babak Saremi, Andreas Beineke, and Klaus Jung. 2022. Augmentation of Transcriptomic Data for Improved Classification of Patients with Respiratory Diseases of Viral Origin. *International journal of molecular sciences* 23, 5 (2022), 2481.

[34] Felix Last, Georgios Douzas, and Fernando Bacao. 2017. Oversampling for imbalanced learning based on k-means and smote. *arXiv preprint arXiv:1711.00837* (2017).

[35] Daniel D Lee and H Sebastian Seung. 1999. Learning the parts of objects by non-negative matrix factorization. *Nature* 401, 6755 (1999), 788–791.

[36] Guillaume Lemaître, Fernando Nogueira, and Christos K. Aridas. 2017. Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning. *Journal of Machine Learning Research* 18, 17 (2017), 1–5. http://jmlr.org/papers/v18/16-365.html

[37] Junnan Li, Qingsheng Zhu, Quanwang Wu, and Zhu Fan. 2021. A novel oversampling technique for class-imbalanced learning based on SMOTE and natural neighbors. *Information Sciences* 565 (2021), 438–455.

[38] Michael I Love, Wolfgang Huber, and Simon Anders. 2014. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome biology* 15, 12 (2014), 1–21.

[39] Polina Mamoshina, Armando Vieira, Evgeny Putin, and Alex Zhavoronkov. 2016. Applications of deep learning in biomedicine. *Molecular pharmaceutics* 13, 5 (2016), 1445–1454.

[40] John C Marioni, Christopher E Mason, Shrikant M Mane, Matthew Stephens, and Yoav Gilad. 2008. RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome research* 18, 9 (2008), 1509–1517.

[41] Mohamed Marouf, Pierre Machart, Vikas Bansal, Christoph Kilian, Daniel S Magruder, Christian F Krebs, and Stefan Bonn. 2020. Realistic in silico generation and augmentation of single-cell RNA-seq data using generative adversarial networks. *Nature communications* 11, 1 (2020), 1–12.

[42] Leland McInnes, John Healy, and James Melville. 2018. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426* (2018).

[43] Radhika Khetani Meeta Mistry. 2020. Gene-level differential expression analysis. https://hbctraining.github.io/DGE_workshop_salmon_online/lessons/01c_RNAseq_count_distribution.html

[44] Byung-Hoon Min, Jeong Mo Bae, Eui Jin Lee, Hong Suk Yu, Young-Ho Kim, Dong Kyung Chang, Hee Cheol Kim, Cheol Keun Park, Suk-Hee Lee, Kyoung-Mee Kim, et al. 2011. The CpG island methylator phenotype may confer a survival benefit in patients with stage II or III colorectal carcinomas receiving fluoropyrimidine-based adjuvant chemotherapy. *BMC cancer* 11, 1 (2011), 1–10.

[45] Cancer Genome Atlas Network et al. 2012. Comprehensive molecular characterization of human colon and rectal cancer. *Nature* 487, 7407 (2012), 330.

[46] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[47] S Popat, R Hubner, and RS Houlston. 2005. Systematic review of microsatellite instability and colorectal cancer prognosis. *Journal of clinical oncology* 23, 3 (2005), 609–618.

[48] Alvin Rajkomar, Jeffrey Dean, and Isaac Kohane. 2019. Machine learning in medicine. *New England Journal of Medicine* 380, 14 (2019), 1347–1358.

[49] Mark D Robinson, Davis J McCarthy, and Gordon K Smyth. 2010. edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *bioinformatics* 26, 1 (2010), 139–140.

[50] Nuria Rodriguez-Salas, Gema Dominguez, Rodrigo Barderas, Marta Mendiola, Xabier García-Albéniz, Juan Maurel, and Jaime Feliu Batlle. 2017. Clinical relevance of colorectal cancer molecular subtypes. *Critical reviews in oncology/hematology* 109 (2017), 9–19.

[51] Yvan Saeys, Inaki Inza, and Pedro Larranaga. 2007. A review of feature selection techniques in bioinformatics. *bioinformatics* 23, 19 (2007), 2507–2517.

[52] Thomas Schaffter, Daniel Marbach, and Dario Floreano. 2011. GeneNetWeaver: in silico benchmark generation and performance profiling of network inference methods. *Bioinformatics* 27, 16 (2011), 2263–2270.

[53] Stacey Shiovitz, Monica M Bertagnolli, Lindsay A Renfro, Eunmi Nam, Nathan R Foster, Slavomir Dzieciatkowski, Yanxin Luo, Victoria Valinluck Lao, Raymond J Monnat Jr, Mary J Emond, et al. 2014. CpG island methylator phenotype is associated with response to adjuvant irinotecan-based therapy for stage III colon cancer. *Gastroenterology* 147, 3 (2014), 637–645.

[54] Connor Shorten and Taghi M Khoshgoftaar. 2019. A survey on image data augmentation for deep learning. *Journal of big data* 6, 1 (2019), 1–48.

[55] TO Sillo, AD Beggs, DG Morton, and G Middleton. 2019. Mechanisms of immunogenicity in colorectal cancer. *Journal of British Surgery* 106, 10 (2019), 1283–1297.

[56] Andrew I Su, John B Welsh, Lisa M Sapinoso, Suzanne G Kern, Petre Dimitrov, Hilmar Lapp, Peter G Schultz, Steven M Powell, Christopher A Moskaluk, Henry F Frierson Jr, et al. 2001. Molecular classification of human carcinomas by use of gene expression signatures. *Cancer research* 61, 20 (2001), 7388–7393.

[57] Masashi Sugiyama and Motoaki Kawanabe. 2012. *Machine learning in non-stationary environments: Introduction to covariate shift adaptation.* MIT press.
[58] Tim Van den Bulcke, Koenraad Van Leemput, Bart Naudts, Piet van Remortel, Hongwu Ma, Alain Verschoren, Bart De Moor, and Kathleen Marchal. 2006. SynTReN: a generator of synthetic gene expression data for design and analysis of structure learning algorithms. *BMC bioinformatics* 7, 1 (2006), 1–12.
[59] Ramon Viñas, Helena Andrés-Terré, Pietro Liò, and Kevin Bryson. 2022. Adversarial generation of gene expression data. *Bioinformatics* 38, 3 (2022), 730–737.
[60] Qikang Wan, Xiongshi Deng, Min Li, and Haotian Yang. 2022. SDDSMOTE: Synthetic Minority Oversampling Technique based on Sample Density Distribution for Enhanced Classification on Imbalanced Microarray Data. In *2022 The 6th International Conference on Compute and Data Analysis.* 35–42.
[61] Likun Wang, Zhixing Feng, Xi Wang, Xiaowo Wang, and Xuegong Zhang. 2010. DEGseq: an R package for identifying differentially expressed genes from RNA-seq data. *Bioinformatics* 26, 1 (2010), 136–138.
[62] Ozlem Yersal and Sabri Barutca. 2014. Biological subtypes of breast cancer: Prognostic and therapeutic implications. *World journal of clinical oncology* 5, 3 (2014), 412.
[63] Xu Zhou, Kai Hu, Peter Bailey, Christoph Springfeld, Susanne Roth, Roma Kurilov, Benedikt Brors, Thomas Gress, Malte Buchholz, Jingyu An, et al. 2021. Clinical impact of molecular subtyping of pancreatic cancer. *Frontiers in cell and developmental biology* 9 (2021), 743908.

A Intra-class crossover sampling formalisation

Consider a dataset with n samples x_i, where $i \in 1, ..., n$, and m phenotype variants P_j, where $j \in 1, ..., m$. Each sample x_i is of one phenotype P_j. Given each variant P_j is associated with a gene signature S_j, we represent each sample by the set of signatures associated with all possible variants, that is, $x_i = \{S_1, ..., S_m\}$. The number of genes in a signature is given by $|S_j|$. Therefore, each patient is represented by $\sum_{j=1}^{m} |S_j|$ genes. For the sake of simplicity, the genes are ordered by signature - all genes belonging to signature S_1 come first and so on, thus creating signature blocks, one for each phenotypic variant. To generate a new sample, x_{n+1} with phenotype $P_{j=1}$, we first subset all samples, belonging to this phenotype (Equation 3). This subset, $X_{j=1}$, is analogous to a bag of signature blocks (Equation 4).

$$X_{j=1} = \{x_i : x_i \in P_{j=1} \forall i \in 1, ..., n\} \tag{3}$$

$$X_{j=1} = \{S_{ij} : j \in 1, ..., m \forall i : x_i \in P_{j=1}\} \tag{4}$$

We randomly sample signature blocks from this subset (Equation 5) which are put together to create a new sample x_{n+1}.

$$x_{n+1} = \{S_j \in_R X_{j=1} \forall j \in 1, ..., m\} \tag{5}$$

We illustrate a practical example on the gene expression of patients with colorectal cancer, using the Consensus Molecular Subtype (CMS) as phenotype in subsubsection 3.1.1.

B Inter-class crossover sampling formalisation

Consider a similar premise as the one described in subsubsection 3.1.1. To generate a synthetic sample of phenotype $P_{j=1}$, we sample $S_{j=1}$ only from the subset of patients, $X_{j=1}$, that have phenotype $P_{j=1}$, that is. $S_j \in_R X_j$. We then randomly sample the remainder signatures S_k for $k \in 1, ..., m$ and $k \neq j$, from all samples except those that belong to phenotype P_k as described in Equation 6. The sampled signatures are then put together to create a new synthetic sample, $x_{n+1} = \{S_j, S_k\}$ for $k \in 1, ..., m$ and $k \neq j$ and where $j = 1$.

$$S_k \in_R \{X - X_k\} \; \forall k \in 1, ..., m \text{ and } k \neq j \tag{6}$$

C Modified gamma-Poisson sampling

The negative-binomial (NB) distribution is a commonly used parametric method for augmenting RNA-Seq data for differential expression analysis due to it's ability to capture the overdispersion typically seen in these data. To model the RNA-Seq read frequency Y of a given gene j, the distribution relies on two parameters, namely the mean and dispersion of the counts, i.e, $Y_j = \text{NB}(\mu_j, \phi)$.

Here, ϕ is the dispersion parameter that controls the variance of the counts, $Var(Y_j) = \mu_j + \phi\mu_j^2$. The estimation of ϕ is, however, not trivial. The negative-binomial (NB) distribution is typically expressed as a gamma-Poisson mixture [2, 20] to make it more tractable. In this formulation, the rate parameter λ of the Poisson distribution is a random variable sampled from a gamma distribution parameterised by the shape (α) and rate (β) parameters, which are estimated from the observed population, $\lambda \sim \Gamma(\alpha, \beta)$. The RNA-Seq read counts are then sampled from this initialised Poisson distribution, $Y_j \sim Poisson(\lambda_j)$.

To ensure that the newly generated samples add some diversity while maintaining the same distribution as the augmented class, we create a mixture of gamma-Poisson distributions. In this strategy, to generate every new observation, a subset S of samples is randomly chosen to estimate the α and β parameters from its mean μ and variance σ^2 as shown in Equation 1. These parameters initialise the gamma distribution. Since the augmentation is aimed at solving class imbalance, the subset is created from samples belonging to the same class. Thus, the newly generated observation will have the same label as the subset. The size of the subset $|S|$ is a hyperparameter defined by the user. The random variable sampled from this gamma distribution is used to initialise the Poisson distribution from which a new observation is sampled. This process is repeated n times to generate n new observations. We recommend a smaller subset size to maximise the variance in the subset. A larger variance would mean more scattered observations, thereby generating a diverse set of samples as opposed to dense local clusters with a smaller variance.

D Modified Poisson sampling

The Poisson distribution is another method commonly used to model RNA-Seq read counts. Similarly to the gamma-Poisson strategy, a subset is first sampled from the class to be augmented, after which the mean is computed and set as the rate parameter λ for the Poisson distribution as shown in Equation 2.

However, unlike the gamma-Poisson, this distribution assumes the variance of the sample equals the mean and fails to capture overdispersion in the data. To generate new observations with this method, a larger subset size is recommended such that the variance is closer to the mean while still being high enough to generate variability in the new data.

E Visualising augmented data for other class sizes

Figure 10 illustrates the UMAP projected distributions of the datasets augmented by various methods to class size 5000. These visualisations are compared to the UMAP projection of the unaugmented dataset to ascertain visually if global and local trends are fairly

Table 2: Sample counts of the different colorectal cancer subtypes in the TCGA and CPTAC gene expression datasets.

Datasets	CMS1	CMS2	CMS3	CMS4	Total
TCGA	74	216	74	142	506
CPTAC	14	33	16	22	85

captured. Similar visualisation are also show for class size Max in Figure 8 and class size 500 in Figure 9.

F Distribution size effect

In this experiment, we test various reference set sizes to understand its impact on CMS classification performance. The reference set sizes considered are $r = 3, 5, 10, 25, 50$. We choose $class_size = 5000$ so that there are enough newly sampled observations to draw a conclusion about the effect of reference set size. The study is conducted on a classification task with various classifiers and the 5-fold cross validation balanced accuracy is reported in Table 3. The reference set size with the highest accuracy on average, $r = 5$, is chosen for the gamma-Poisson sampling method. Figure 11 illustrates what the distribution looks like with different reference set sizes. We observe that for $r = 5$, the real and synthetic samples are well interspersed with each other showing qualitatively that the distribution of synthetic samples is similar to that of the real samples. As r increases, new samples are created that seemingly fall away from its member class resulting in new sub-clusters as seen in Figure 11 c and d, for CMS1 and CMS2 classes.

For the Poisson distribution, we double the reference set size to $r = 10$ such that the variance is more stable and closer to the mean. Figure 12 illustrates the Poisson augmented data at different reference set sizes. Here, we make a similar observation that as r increases, sub-clusters begin to form due to synthetic samples being generated well outside its member class distribution. On comparing the Poisson augmented data to the gamma-Poisson augmented data, we observe that the CMS clusters appear to be well separated with the Poisson. While this may appear to be beneficial for certain tasks that are strictly associated with the these types, it does not capture the distribution of the real data well (Figure 10a). This could introduce spurious associations and attributions.

While we acknowledge that conclusions drawn from UMAP visualisations have to be taken with a grain of salt, we note that all UMAPs were generated with the same hyperparameter choice, differing only in their input data. This gives us a qualitative insight into what these differently augmented datasets with various reference set sizes look like and whether or not the newly generated data improve the support of a given class.

We omit results for $r = 3$ since it has high failure rates. Since the parametric distributions depend on the mean and variance of the subsample, it can sometimes happen that when $r = 3$ the variance associated with the subsample is too small, resulting in NaNs being sampled from the gamma distribution due to an invalid argument (negative integers or 0 possibly due to a possible overflow error).

Table 3: Effect of reference sample size on classifier performance for parametric methods. The table shows 5-fold cross-validation results for different r sizes and classifiers. Majority of the best performances belong to $r = 5$, which also has the best performance on average.

Model	$r = 5$	$r = 10$	$r = 25$	$r = 50$
KNN	0.857 ± 0.035	0.836 ± 0.027	0.836 ± 0.036	**0.858 ± 0.029**
LR	**0.889 ± 0.031**	0.881 ± 0.040	0.886 ± 0.033	0.888 ± 0.049
SVM	0.886 ± 0.027	**0.893 ± 0.033**	0.889 ± 0.027	0.887 ± 0.036
RF	**0.881 ± 0.029**	0.865 ± 0.036	0.857 ± 0.037	0.859 ± 0.035
EBM	**0.905 ± 0.011**	0.892 ± 0.028	0.881 ± 0.032	0.882 ± 0.030
MLP	0.881 ± 0.020	0.889 ± 0.040	**0.889 ± 0.025**	0.883 ± 0.038

G Implementation Details

G.1 Classification Performance

To demonstrate the utility of our augmentation methods in discriminative modelling, we make use of 5 classifiers, namely, Logistic Regression (LR), K-Nearest Neighbours (KNN), Support Vector Machines (SVM), Explainable Boosting Machines (EBM), and Random Forests (RF). For the standard classifiers, we make use of the `scikit-learn` [46] library implementations with the default parameters. We only increase the `max_iter` to 5000 for the Logistic Regression model. For the EBMs, we use the `InterpretML` library. We set a seed (42) for all models.

We report the mean balanced accuracy (Equation 7) on the TCGA standardised test sets and the CPTAC dataset. The metric is computed using true positive (TP), true negative (TN), false positive (FP) and false negative (FN) counts and is averaged across the 25 splits (5-fold CV repeated 5 times). The CPTAC dataset is standardised according to the training data mean and standard deviation from each split after which it is passed to the trained model from that split. We drop patients for whom the target label is not available. Additionally, we perform significance testing with the Wilcoxon test. All experiments are run locally on a 2019 MacBook Pro.

$$balanced - accuracy = \frac{1}{2}\left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP}\right) \quad (7)$$

G.1.1 Effect of Data Size. The above process is repeated for augmented data containing various percentages of real data from 10 to 50 percent and 100 percent.

G.1.2 Predicting Other Variables. For the classification tasks of predicting MSI and CIMP status, we use the SVM-RBF and EBM models from `scikit-learn` and `InterpretML` to run our experiments (with the default parameters and seed 42). Here the input to the models is a 4D latent representation extracted from the VAE (described in subsection G.2) for each patient. Similar to the survival analysis task, we perform training and testing with the same 5x5 stratified cross-validation splits, and all samples for which the MSI/CIMP status is not available are dropped for the MSI/CIMP prediction task respectively. The patient summary counts for both MSI and CIMP are reported in Table 4.

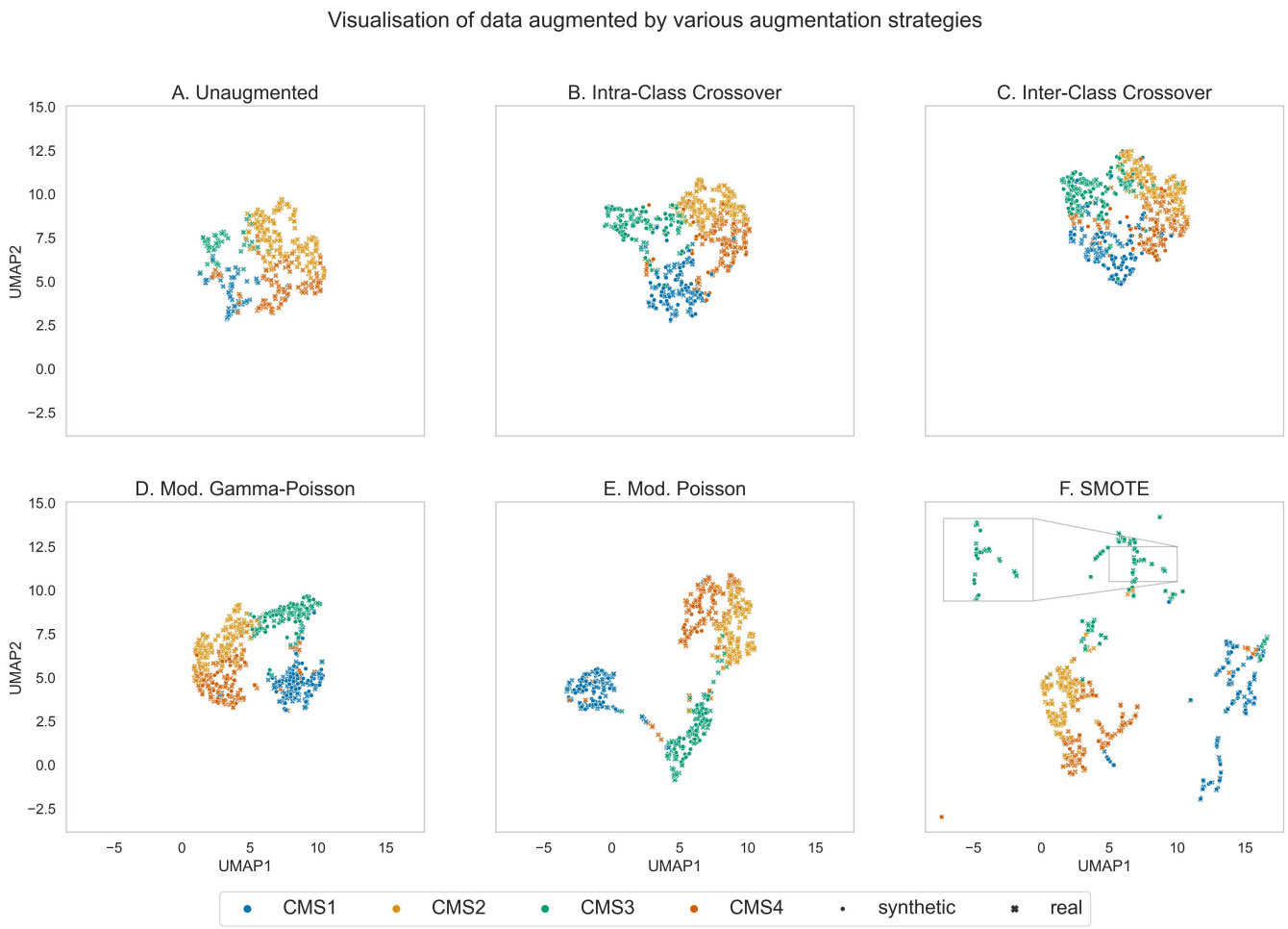

Figure 8: UMAP [42] visualisations of datasets augmented to match the majority class size. The intra-class and inter-class crossover methods capture both the global and local trends in the original unaugmented data better than the other methods. Modified Gamma-Poisson and SMOTE closely follow. The modified Poisson sampling method begins to separate clusters which become more apparent and amplified as the class augmentation size is increased.

Table 4: Summary counts of patients with MSI and CIMP labels available in the TCGA COADREAD and CPTAC COAD gene expression datasets.

Clinical Variable	Categories	TCGA	CPTAC
MSI	MSI-H	80	24
	MSI-L	74	-
	MSS	339	81
	Total	493	105
CIMP	CIMP.High	66	-
	CIMP.Low	86	-
	CIMP.Neg	272	-
	Total	364	-

G.2 Generation Quality

The encoder and decoder of the VAE are single-layer neural networks with 20 hidden units activated by the ReLU function [1]. The dimension of the latent representation is 4. The batch size and learning rate are adapted according to the size of the dataset. For the unaugmented dataset and for the augmented datasets with class sizes 'Max' and '500', we set the batch size to 64 and learning rate to 0.001. For all the other datasets, we set the batch size to 128 and learning rate to 0.0001. The models are trained on the sum of the L2 loss and KL-divergence. An L2 penalty of 0.0001 is applied to all models, which are optimised by Adam [31] and trained for 300 epochs with an early stopping wait time of 8 steps. The experiments to train a VAE were run on CPUs with 24 cores. The experiment for generating new samples and classifying them were run on 12 core CPUs.

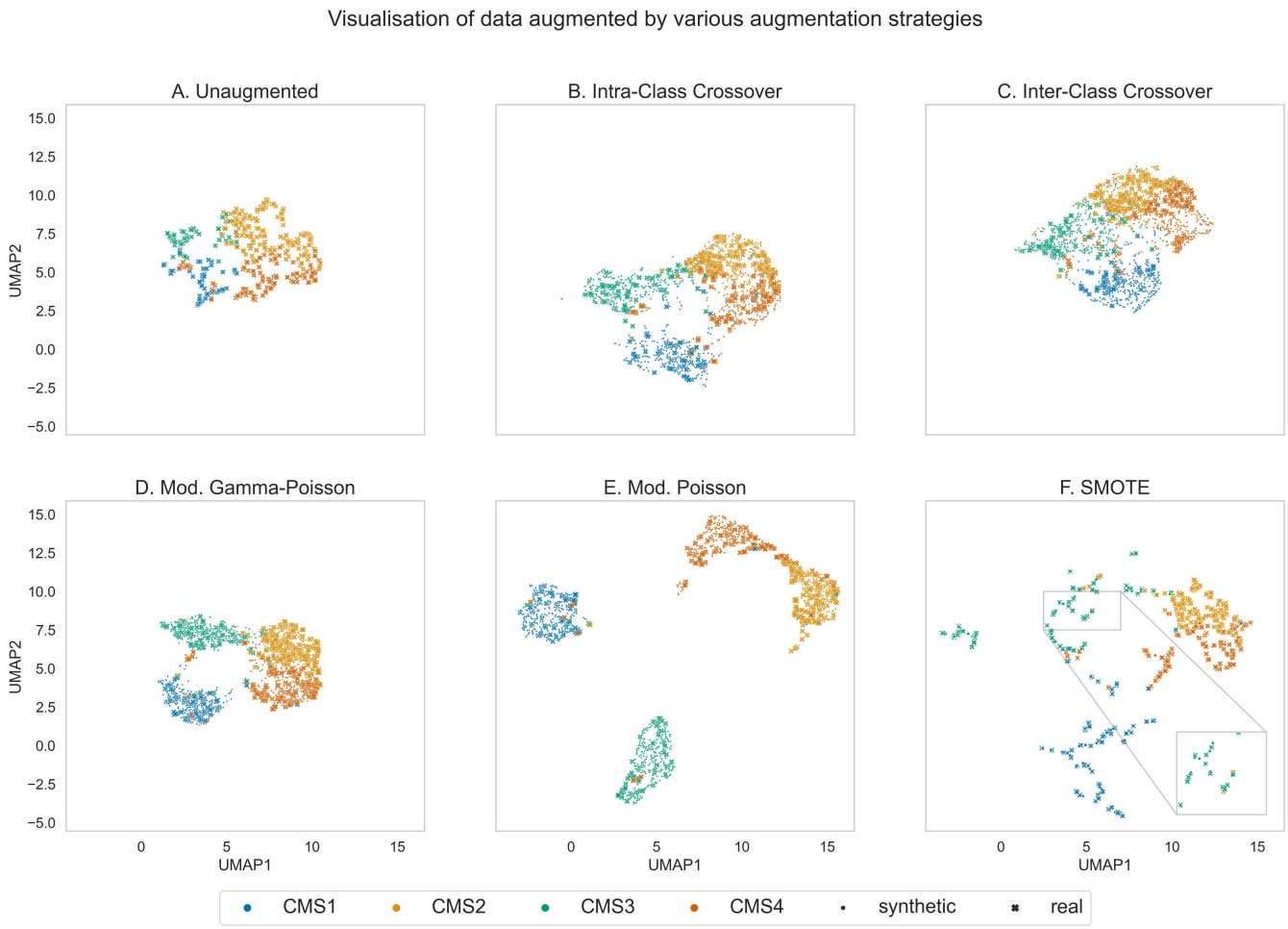

Figure 9: UMAP [42] visualisations of datasets augmented to have 500 samples per class. Data augmented by the intra-class, inter-class crossover methods and Modified Gamma-Poisson sampling maintain a structure similar to the original unaugmented data. Data generated by the modified Poisson sampling method begins to show cluster separation, which could lead to spurious results. SMOTE generates samples on the line between two reference points and thus cannot populate sub-regions densely.

H Extended Results

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009

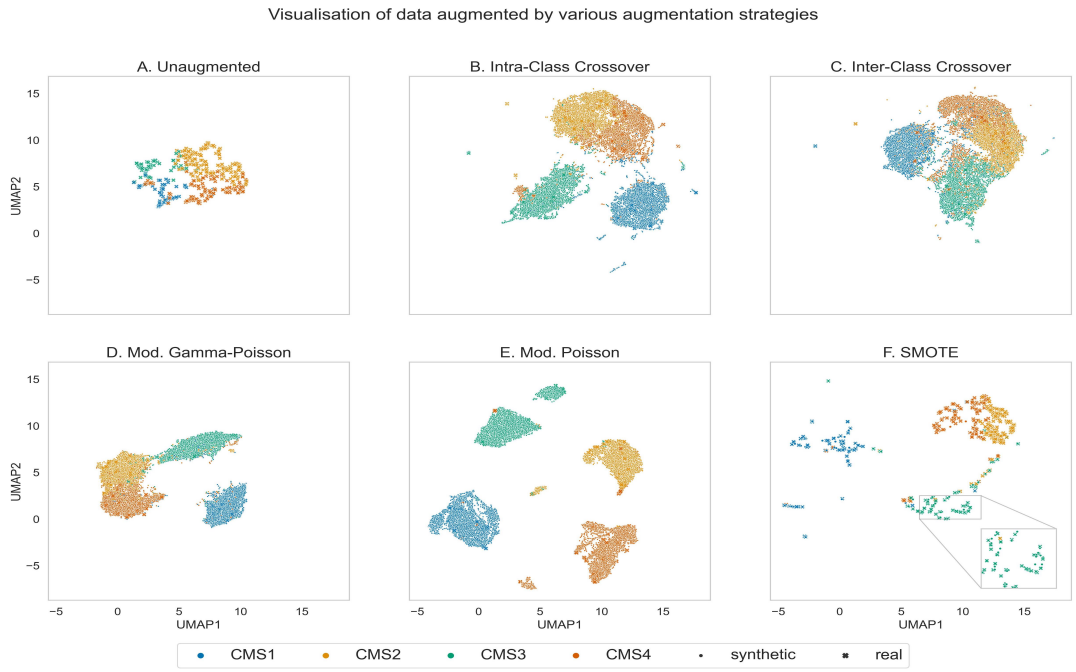

Figure 10: UMAP [42] visualisations of the augmented datasets show that the inter-class crossover sampling method best captures both the global and local structure of the original unaugmented data. All augmented datasets have 5000 samples per class.

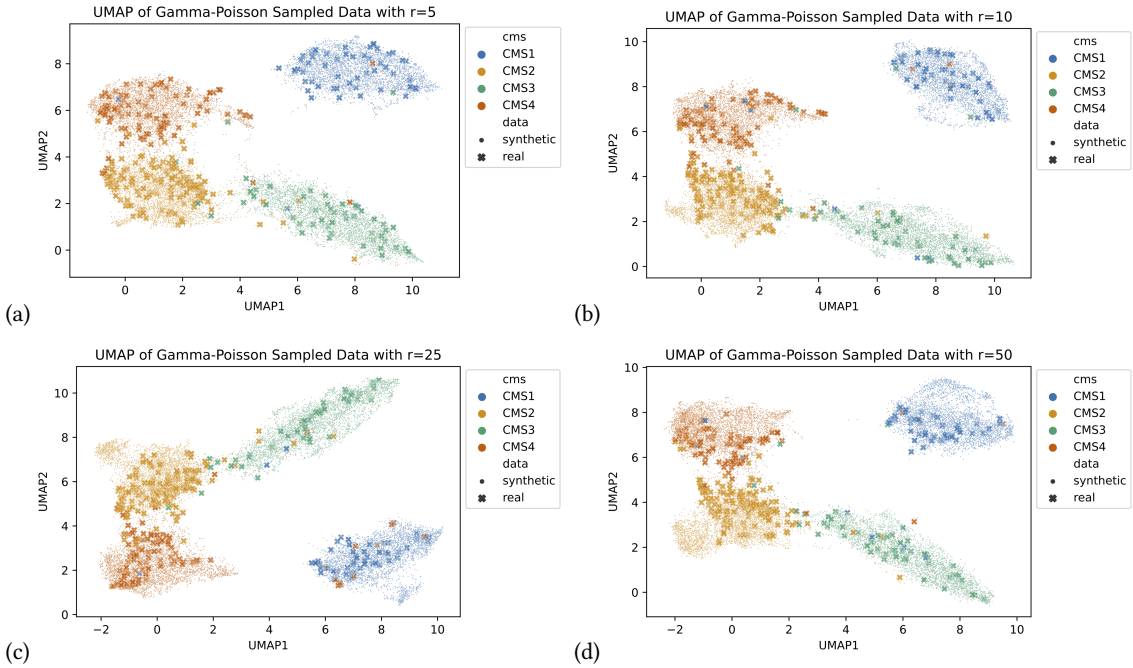

Figure 11: UMAP [42] visualisations of gamma-Poisson augmented datasets in the study. (a)r=5. (b) r=10. (c) r=25. (d) r=50. All augmented datasets have 5000 samples per class. When r is set to 5, the new samples are more interspersed with the real data, as opposed to r=50, where new samples are generated "out of distribution" resulting in new clusters.

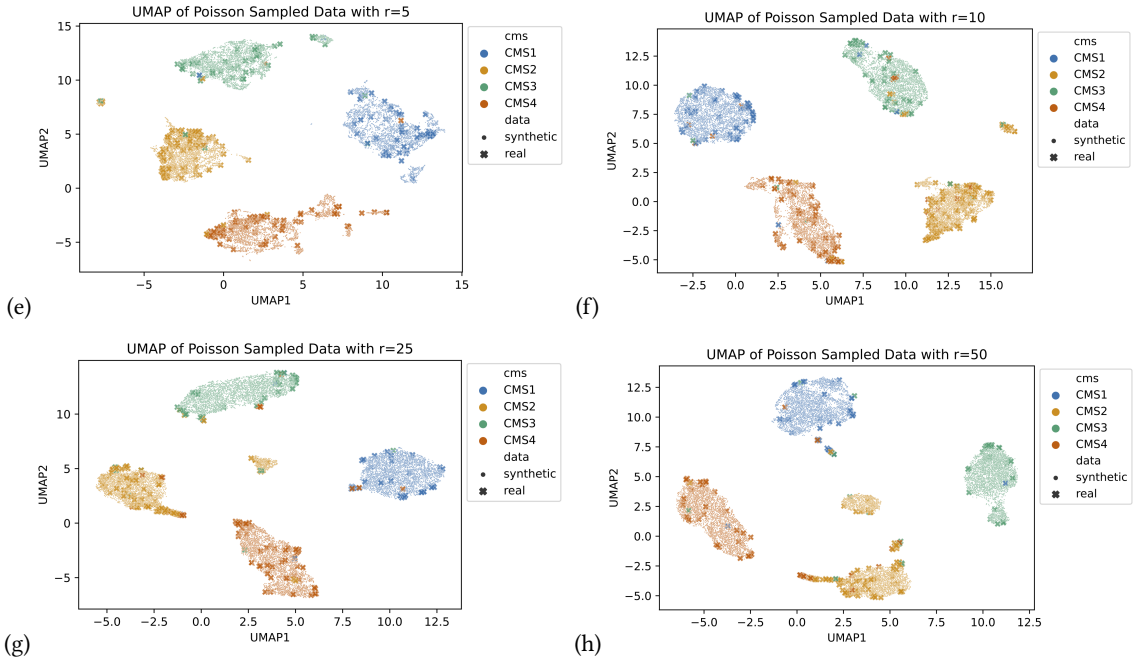

Figure 12: UMAP [42] visualisations of Poisson augmented datasets in the study. (a)r=5. (b) r=10. (c) r=25. (d) r=50. All augmented datasets have 5000 samples per class. For r=10, the new samples appear better interspersed with the real data and there are fewer outlier clusters.

Table 5: Average balanced accuracy on in-domain TCGA test set from 5x5 cross validation with different classifier models and an overall average of these models for different augmentation methods. The entire training dataset (100%) was used for augmentation.

Class Size	Sampling Methods	EBM	KNN	Logistic	RF	SVM-RBF	Average
Max	Mod. Gamma-Poisson	0.8841 ± 0.0297	0.8331 ± 0.0325	0.8936 ± 0.0266	0.8782 ± 0.0316	0.8896 ± 0.0253	0.8757 ± 0.0291
	Inter-Crossover (Ours)	0.8948 ± 0.0276	0.8465 ± 0.0371	0.8917 ± 0.0292	0.8821 ± 0.0344	0.9011 ± 0.0296	0.8832 ± 0.0316
	Intra-Crossover (Ours)	0.8796 ± 0.0244	0.8434 ± 0.0353	0.8837 ± 0.0228	0.8763 ± 0.0317	0.8808 ± 0.0327	0.8728 ± 0.0294
	Mod. Poisson	0.8819 ± 0.0249	0.878 ± 0.0335	0.8885 ± 0.0313	0.8729 ± 0.0377	0.8829 ± 0.0308	0.8808 ± 0.0316
	Replacement	0.8744 ± 0.0328	0.8404 ± 0.04	0.891 ± 0.022	0.8744 ± 0.0288	0.8974 ± 0.0253	0.8755 ± 0.0298
	SMOTE	0.8904 ± 0.0225	0.8666 ± 0.0342	0.8852 ± 0.0306	0.8725 ± 0.0306	0.8965 ± 0.0243	0.8822 ± 0.0284
	Unaugmented	0.8674 ± 0.0264	0.841 ± 0.0354	0.8889 ± 0.0319	0.8615 ± 0.0335	0.8861 ± 0.0315	0.869 ± 0.0317
500	Mod. Gamma-Poisson	0.9003 ± 0.0285	0.8364 ± 0.0368	0.8991 ± 0.0217	0.8777 ± 0.0284	0.8955 ± 0.0264	0.8818 ± 0.0283
	Inter-Crossover (Ours)	0.8762 ± 0.0334	0.8319 ± 0.041	0.8765 ± 0.0266	0.8727 ± 0.0275	0.8846 ± 0.0314	0.8684 ± 0.032
	Intra-Crossover (Ours)	0.8827 ± 0.0299	0.8485 ± 0.0408	0.8763 ± 0.0245	0.8704 ± 0.0379	0.8873 ± 0.0298	0.873 ± 0.0326
	Mod. Poisson	0.8825 ± 0.027	0.8739 ± 0.0402	0.8858 ± 0.027	0.8651 ± 0.0322	0.885 ± 0.0285	0.8784 ± 0.031
	Replacement	0.8684 ± 0.0347	0.8226 ± 0.0477	0.8792 ± 0.0297	0.8728 ± 0.0322	0.8887 ± 0.0309	0.8663 ± 0.035
	SMOTE	0.8877 ± 0.0264	0.8602 ± 0.0406	0.888 ± 0.0315	0.8711 ± 0.0324	0.8938 ± 0.0261	0.8801 ± 0.0314
	Unaugmented	0.8674 ± 0.0264	0.841 ± 0.0354	0.8889 ± 0.0319	0.8615 ± 0.0335	0.8861 ± 0.0315	0.869 ± 0.0317
5000	Mod. Gamma-Poisson	0.8927 ± 0.0291	0.8583 ± 0.0356	0.8955 ± 0.0256	0.8781 ± 0.0353	0.897 ± 0.0238	0.8843 ± 0.0299
	Inter-Crossover (Ours)	0.8445 ± 0.0351	0.837 ± 0.0403	0.8664 ± 0.0298	0.8479 ± 0.0378	0.8649 ± 0.0297	0.8521 ± 0.0345
	Intra-Crossover (Ours)	0.8603 ± 0.0281	0.8472 ± 0.0465	0.8668 ± 0.0278	0.8636 ± 0.0377	0.8723 ± 0.0257	0.8621 ± 0.0332
	Mod. Poisson	0.8832 ± 0.0282	0.8688 ± 0.0383	0.8939 ± 0.0276	0.8674 ± 0.0352	0.8893 ± 0.0272	0.8805 ± 0.0313
	Replacement	0.8654 ± 0.0346	0.8048 ± 0.045	0.8669 ± 0.0349	0.8622 ± 0.0374	0.8849 ± 0.0267	0.8568 ± 0.0357
	SMOTE	0.8812 ± 0.033	0.8464 ± 0.0405	0.8776 ± 0.0296	0.8668 ± 0.0346	0.8927 ± 0.0292	0.8729 ± 0.0334
	Unaugmented	0.8674 ± 0.0264	0.841 ± 0.0354	0.8889 ± 0.0319	0.8615 ± 0.0335	0.8861 ± 0.0315	0.869 ± 0.0317

Table 6: Average balanced accuracy on out-of-domain CPTAC test set from 5x5 cross validation with different classifier models and an overall average of these models for different augmentation methods. The entire training dataset (100%) was used for augmentation.

Class Size	Sampling Methods	EBM	KNN	Logistic	RF	SVM-RBF	Average
Max	Mod. Gamma-Poisson	0.6506 ± 0.0334	0.5781 ± 0.0297	0.5396 ± 0.0513	0.6348 ± 0.0248	0.6025 ± 0.029	0.6011 ± 0.0336
	Inter-Crossover (Ours)	0.67 ± 0.0342	0.5876 ± 0.036	0.5179 ± 0.0523	0.6306 ± 0.0181	0.599 ± 0.0268	0.601 ± 0.0335
	Intra-Crossover (Ours)	0.6474 ± 0.0299	0.5947 ± 0.0374	0.509 ± 0.0501	0.6436 ± 0.0193	0.5986 ± 0.0253	0.5986 ± 0.0324
	Mod. Poisson	0.6332 ± 0.0318	0.6225 ± 0.026	0.537 ± 0.0432	0.6309 ± 0.0401	0.6013 ± 0.0372	0.605 ± 0.0357
	Replacement	0.6314 ± 0.0331	0.5822 ± 0.0311	0.55 ± 0.0302	0.6369 ± 0.032	0.5838 ± 0.0367	0.5969 ± 0.0326
	SMOTE	0.6375 ± 0.0353	0.5961 ± 0.0333	0.5361 ± 0.0323	0.6557 ± 0.0274	0.5788 ± 0.0347	0.6009 ± 0.0326
	Unaugmented	0.637 ± 0.0252	0.5917 ± 0.0257	0.5347 ± 0.0337	0.6256 ± 0.0354	0.6058 ± 0.0285	0.599 ± 0.0297
500	Mod. Gamma-Poisson	0.6732 ± 0.0334	0.6 ± 0.0346	0.5681 ± 0.0485	0.654 ± 0.0196	0.6055 ± 0.0313	0.6202 ± 0.0335
	Inter-Crossover (Ours)	0.7135 ± 0.0242	0.6359 ± 0.034	0.6213 ± 0.0346	0.6657 ± 0.0329	0.6303 ± 0.0345	0.6534 ± 0.032
	Intra-Crossover (Ours)	0.6705 ± 0.0281	0.6145 ± 0.0355	0.5646 ± 0.0475	0.6771 ± 0.0206	0.6017 ± 0.0253	0.6257 ± 0.0314
	Mod. Poisson	0.6256 ± 0.0314	0.6402 ± 0.03	0.5313 ± 0.026	0.6353 ± 0.033	0.5892 ± 0.0418	0.6043 ± 0.0324
	Replacement	0.6255 ± 0.0261	0.5732 ± 0.036	0.5531 ± 0.0361	0.6275 ± 0.024	0.5673 ± 0.0343	0.5893 ± 0.0313
	SMOTE	0.616 ± 0.0343	0.6009 ± 0.0378	0.5339 ± 0.0268	0.6491 ± 0.0309	0.5642 ± 0.0344	0.5928 ± 0.0328
	Unaugmented	0.637 ± 0.0252	0.5917 ± 0.0257	0.5347 ± 0.0337	0.6256 ± 0.0354	0.6058 ± 0.0285	0.599 ± 0.0297
5000	Mod. Gamma-Poisson	0.6845 ± 0.0145	0.6253 ± 0.0287	0.6344 ± 0.0197	0.6775 ± 0.0131	0.6243 ± 0.0284	0.6492 ± 0.0209
	Inter-Crossover (Ours)	0.6532 ± 0.0183	0.6748 ± 0.0253	0.6533 ± 0.0153	0.6821 ± 0.0208	0.6592 ± 0.0154	0.6645 ± 0.019
	Intra-Crossover (Ours)	0.6637 ± 0.0421	0.6367 ± 0.0299	0.5917 ± 0.038	0.673 ± 0.023	0.605 ± 0.0279	0.634 ± 0.0322
	Mod. Poisson	0.6243 ± 0.0222	0.6354 ± 0.0192	0.5293 ± 0.023	0.6057 ± 0.0252	0.571 ± 0.0358	0.5932 ± 0.0251
	Replacement	0.6271 ± 0.0353	0.5725 ± 0.0276	0.5588 ± 0.0518	0.6206 ± 0.0269	0.58 ± 0.0374	0.5918 ± 0.0358
	SMOTE	0.6228 ± 0.0325	0.5842 ± 0.0274	0.5336 ± 0.0412	0.6139 ± 0.0324	0.5709 ± 0.041	0.5851 ± 0.0349
	Unaugmented	0.637 ± 0.0252	0.5917 ± 0.0257	0.5347 ± 0.0337	0.6256 ± 0.0354	0.6058 ± 0.0285	0.599 ± 0.0297

Table 7: Average balanced accuracy on in-domain TCGA test set from 5x5 cross validation with different classifier models and an overall average of these models for different augmentation methods. A portion of the training dataset (10%) was used for augmentation.

Class Size	Sampling Methods	EBM	KNN	Logistic	RF	SVM-RBF	Average
Max	Mod. Gamma-Poisson	0.8126 ± 0.0455	0.7525 ± 0.0542	0.8363 ± 0.0609	0.7914 ± 0.0684	0.8182 ± 0.0567	0.8022 ± 0.0572
	Inter-Crossover (Ours)	0.8092 ± 0.0536	0.7666 ± 0.0463	0.8349 ± 0.058	0.8129 ± 0.0433	0.8258 ± 0.0581	0.8099 ± 0.0518
	Intra-Crossover (Ours)	0.8141 ± 0.0524	0.7697 ± 0.0535	0.8272 ± 0.0589	0.7973 ± 0.0568	0.822 ± 0.0554	0.8061 ± 0.0554
	Mod. Poisson	0.8049 ± 0.0593	0.8191 ± 0.0418	0.8323 ± 0.0564	0.7838 ± 0.0682	0.7965 ± 0.0593	0.8073 ± 0.057
	Replacement	0.7983 ± 0.052	0.7467 ± 0.0628	0.836 ± 0.0566	0.7842 ± 0.0533	0.8097 ± 0.0563	0.795 ± 0.0562
	SMOTE	0.806 ± 0.0513	0.7827 ± 0.0534	0.835 ± 0.0559	0.7918 ± 0.0659	0.8203 ± 0.0562	0.8072 ± 0.0566
	Unaugmented	0.7707 ± 0.0551	0.7494 ± 0.062	0.8165 ± 0.0561	0.7858 ± 0.047	0.7863 ± 0.0509	0.7817 ± 0.0542
500	Mod. Gamma-Poisson	0.8341 ± 0.0497	0.8012 ± 0.0521	0.8433 ± 0.0498	0.8023 ± 0.0633	0.8412 ± 0.0492	0.8244 ± 0.0528
	Inter-Crossover (Ours)	0.789 ± 0.051	0.7889 ± 0.0414	0.7867 ± 0.0522	0.7851 ± 0.0483	0.8116 ± 0.0483	0.7923 ± 0.0482
	Intra-Crossover (Ours)	0.7824 ± 0.0479	0.7856 ± 0.0519	0.7974 ± 0.0488	0.7521 ± 0.0571	0.7849 ± 0.0519	0.7805 ± 0.0515
	Mod. Poisson	0.822 ± 0.0589	0.8076 ± 0.0542	0.8463 ± 0.0449	0.7724 ± 0.0533	0.6977 ± 0.0999	0.7892 ± 0.0622
	Replacement	0.8009 ± 0.0527	0.7399 ± 0.0625	0.8336 ± 0.0521	0.7653 ± 0.0572	0.8118 ± 0.0614	0.7903 ± 0.0572
	SMOTE	0.8209 ± 0.0568	0.8036 ± 0.0577	0.8403 ± 0.0597	0.7825 ± 0.0623	0.8232 ± 0.0553	0.8141 ± 0.0584
	Unaugmented	0.7707 ± 0.0551	0.7494 ± 0.062	0.8165 ± 0.0561	0.7858 ± 0.047	0.7863 ± 0.0509	0.7817 ± 0.0542
5000	Mod. Gamma-Poisson	0.838 ± 0.0521	0.8154 ± 0.0505	0.8353 ± 0.0618	0.7972 ± 0.0661	0.841 ± 0.055	0.8254 ± 0.0571
	Inter-Crossover (Ours)	0.761 ± 0.0529	0.7852 ± 0.0543	0.7574 ± 0.0516	0.7473 ± 0.0553	0.7615 ± 0.0557	0.7625 ± 0.054
	Intra-Crossover (Ours)	0.7661 ± 0.0505	0.7474 ± 0.0514	0.7785 ± 0.057	0.7006 ± 0.0663	0.7672 ± 0.04	0.752 ± 0.053
	Mod. Poisson	0.8221 ± 0.054	0.7969 ± 0.0548	0.8381 ± 0.046	0.7742 ± 0.0508	0.6887 ± 0.1054	0.784 ± 0.0622
	Replacement	0.801 ± 0.0569	0.7407 ± 0.063	0.8273 ± 0.0506	0.7688 ± 0.0631	0.8119 ± 0.0622	0.7899 ± 0.0591
	SMOTE	0.8157 ± 0.0562	0.7994 ± 0.0571	0.835 ± 0.0556	0.7693 ± 0.0581	0.823 ± 0.0567	0.8085 ± 0.0568
	Unaugmented	0.7707 ± 0.0551	0.7494 ± 0.062	0.8165 ± 0.0561	0.7858 ± 0.047	0.7863 ± 0.0509	0.7817 ± 0.0542

Table 8: Average balanced accuracy on out-of-domain CPTAC test set from 5x5 cross validation with different classifier models and an overall average of these models for different augmentation methods. A portion of the training dataset (10%) was used for augmentation.

Class Size	Sampling Methods	EBM	KNN	Logistic	RF	SVM-RBF	Average
Max	Mod. Gamma-Poisson	0.6505 ± 0.054	0.5856 ± 0.0545	0.6016 ± 0.0734	0.6505 ± 0.0449	0.6129 ± 0.0682	0.6202 ± 0.059
	Inter-Crossover (Ours)	0.6682 ± 0.0481	0.5847 ± 0.0645	0.63 ± 0.0743	0.6429 ± 0.0539	0.6321 ± 0.0751	0.6316 ± 0.0632
	Intra-Crossover (Ours)	0.6587 ± 0.0446	0.6145 ± 0.0642	0.6173 ± 0.0633	0.6488 ± 0.0548	0.6249 ± 0.0674	0.6328 ± 0.0589
	Mod. Poisson	0.6467 ± 0.0493	0.6182 ± 0.0381	0.628 ± 0.0601	0.6574 ± 0.0718	0.6217 ± 0.0631	0.6344 ± 0.0565
	Replacement	0.6413 ± 0.0487	0.5949 ± 0.062	0.6199 ± 0.0731	0.6434 ± 0.0512	0.6135 ± 0.0766	0.6226 ± 0.0623
	SMOTE	0.6594 ± 0.0465	0.6013 ± 0.0501	0.6348 ± 0.0659	0.6649 ± 0.0563	0.6265 ± 0.0655	0.6374 ± 0.0569
	Unaugmented	0.6278 ± 0.052	0.5464 ± 0.0756	0.5707 ± 0.0739	0.5923 ± 0.0743	0.5848 ± 0.0915	0.5844 ± 0.0735
500	Mod. Gamma-Poisson	0.6904 ± 0.0564	0.6397 ± 0.0521	0.6184 ± 0.0775	0.6454 ± 0.0456	0.6348 ± 0.0539	0.6457 ± 0.0571
	Inter-Crossover (Ours)	0.6638 ± 0.0461	0.6768 ± 0.0447	0.6358 ± 0.0653	0.6521 ± 0.0441	0.6492 ± 0.0456	0.6555 ± 0.0492
	Intra-Crossover (Ours)	0.6408 ± 0.0673	0.5877 ± 0.076	0.5787 ± 0.0824	0.6239 ± 0.0581	0.5675 ± 0.079	0.5997 ± 0.0726
	Mod. Poisson	0.6329 ± 0.0576	0.6511 ± 0.0434	0.645 ± 0.0627	0.649 ± 0.0677	0.5279 ± 0.1103	0.6212 ± 0.0683
	Replacement	0.6469 ± 0.0424	0.5822 ± 0.0534	0.6169 ± 0.0765	0.6488 ± 0.0583	0.6184 ± 0.0699	0.6226 ± 0.0601
	SMOTE	0.6403 ± 0.0496	0.6155 ± 0.0366	0.6291 ± 0.0642	0.6294 ± 0.0679	0.6271 ± 0.0567	0.6283 ± 0.055
	Unaugmented	0.6278 ± 0.052	0.5464 ± 0.0756	0.5707 ± 0.0739	0.5923 ± 0.0743	0.5848 ± 0.0915	0.5844 ± 0.0735
5000	Mod. Gamma-Poisson	0.6973 ± 0.0389	0.6553 ± 0.056	0.629 ± 0.0587	0.6574 ± 0.0521	0.634 ± 0.052	0.6546 ± 0.0515
	Inter-Crossover (Ours)	0.6233 ± 0.0531	0.6551 ± 0.0326	0.6189 ± 0.0723	0.6277 ± 0.0386	0.6382 ± 0.0426	0.6327 ± 0.0479
	Intra-Crossover (Ours)	0.6256 ± 0.0492	0.5586 ± 0.0534	0.5556 ± 0.0965	0.5921 ± 0.0575	0.5619 ± 0.0777	0.5788 ± 0.0669
	Mod. Poisson	0.63 ± 0.0531	0.6494 ± 0.0411	0.6314 ± 0.0601	0.6465 ± 0.0469	0.5286 ± 0.1039	0.6172 ± 0.061
	Replacement	0.6448 ± 0.0453	0.5808 ± 0.0547	0.6131 ± 0.0805	0.64 ± 0.0704	0.6165 ± 0.0703	0.619 ± 0.0642
	SMOTE	0.6454 ± 0.0483	0.6223 ± 0.0366	0.6243 ± 0.0681	0.6165 ± 0.0732	0.6253 ± 0.0599	0.6268 ± 0.0572
	Unaugmented	0.6278 ± 0.052	0.5464 ± 0.0756	0.5707 ± 0.0739	0.5923 ± 0.0743	0.5848 ± 0.0915	0.5844 ± 0.0735

Table 9: Average balanced accuracy of different classifier models on VAE generated samples modelled on in-domain TCGA test set from 5x5 cross validation and an overall average of these models for different augmentation methods. A portion of the real training dataset (10%) was used for augmentation and each class was augmented to size 500.

Sampling Methods	EBM	KNN	Logistic	RF	SVM-RBF	Average
Mod. Gamma-Poisson	0.7794 ± 0.0548	0.7694 ± 0.0504	0.7829 ± 0.0565	0.7689 ± 0.0553	0.7851 ± 0.0508	0.7771 ± 0.0536
Inter-Crossover (Ours)	0.749 ± 0.0558	0.7362 ± 0.058	0.7609 ± 0.0515	0.7524 ± 0.0538	0.7619 ± 0.0465	0.7521 ± 0.0531
Intra-Crossover (Ours)	0.76 ± 0.0524	0.7568 ± 0.0526	0.7877 ± 0.055	0.7677 ± 0.0561	0.7763 ± 0.0578	0.7697 ± 0.0548
Mod. Poisson	0.7322 ± 0.0714	0.7314 ± 0.0611	0.7327 ± 0.0626	0.7237 ± 0.0839	0.7255 ± 0.0681	0.7291 ± 0.0694
Replacement	0.7612 ± 0.0516	0.7396 ± 0.0687	0.7719 ± 0.0565	0.7567 ± 0.0561	0.7623 ± 0.0573	0.7583 ± 0.0580
SMOTE	0.7639 ± 0.0629	0.7468 ± 0.0528	0.7619 ± 0.0599	0.7544 ± 0.0569	0.7585 ± 0.0548	0.7571 ± 0.0575
Unaugmented	0.2517 ± 0.0091	0.2502 ± 0.0122	0.2514 ± 0.0209	0.2504 ± 0.0162	0.25 ± 0.0	0.2507 ± 0.0117

Table 10: Average balanced accuracy of different classifier models on VAE generated samples modelled on out-of-domain CPTAC test set from 5x5 cross validation and an overall average of these models for different augmentation methods. A portion of the real training dataset (10%) was used for augmentation and each class was augmented to size 500.

Sampling Methods	EBM	KNN	Logistic	RF	SVM-RBF	Average
Mod. Gamma-Poisson	0.6223 ± 0.0574	0.6018 ± 0.0785	0.6009 ± 0.059	0.5978 ± 0.0653	0.6157 ± 0.0704	0.6077 ± 0.0661
Inter-Crossover (Ours)	0.649 ± 0.0632	0.6719 ± 0.0657	0.67 ± 0.0622	0.6489 ± 0.0602	0.678 ± 0.0635	0.6636 ± 0.0630
Intra-Crossover (Ours)	0.6373 ± 0.0754	0.638 ± 0.0585	0.6502 ± 0.0624	0.6225 ± 0.0675	0.6448 ± 0.064	0.6386 ± 0.0656
Mod. Poisson	0.6147 ± 0.066	0.6201 ± 0.0657	0.6305 ± 0.0628	0.5892 ± 0.0621	0.6164 ± 0.0569	0.6142 ± 0.0627
Replacement	0.6198 ± 0.0653	0.5848 ± 0.0708	0.6254 ± 0.0525	0.6014 ± 0.0739	0.6147 ± 0.0568	0.6092 ± 0.0649
SMOTE	0.6203 ± 0.0546	0.617 ± 0.0631	0.624 ± 0.0528	0.6094 ± 0.0616	0.6209 ± 0.0583	0.6183 ± 0.0581
Unaugmented	0.2573 ± 0.0153	0.2512 ± 0.0069	0.2524 ± 0.0213	0.2536 ± 0.0162	0.25 ± 0.0	0.2529 ± 0.0119

Table 11: Average balanced accuracy on in-domain TCGA test data of different classifier models in predicting the MSI status from embeddings retrieved from a VAE and an overall average of these models for different augmentation methods. A portion of the real training dataset (10%) was used for augmentation and each class was augmented to size 500.

Sampling Methods	EBM	KNN	Logistic	RF	SVM-RBF	Average
Mod. Gamma-Poisson	0.5723 ± 0.0483	0.5958 ± 0.0472	0.6117 ± 0.0597	0.5895 ± 0.0617	0.5865 ± 0.0713	0.5912 ± 0.0576
Inter-Crossover (Ours)	0.5683 ± 0.0423	0.6184 ± 0.0498	0.6216 ± 0.0667	0.5962 ± 0.0643	0.6187 ± 0.0578	0.6046 ± 0.0562
Intra-Crossover (Ours)	0.5667 ± 0.0538	0.6177 ± 0.0534	0.6222 ± 0.0691	0.5983 ± 0.0587	0.6223 ± 0.062	0.6054 ± 0.0594
Mod. Poisson	0.5736 ± 0.0601	0.5879 ± 0.0677	0.6023 ± 0.0615	0.5887 ± 0.0472	0.5772 ± 0.0555	0.5859 ± 0.0584
Replacement	0.6057 ± 0.0572	0.5872 ± 0.0543	0.6391 ± 0.0482	0.619 ± 0.0467	0.6124 ± 0.06	0.6127 ± 0.0533
SMOTE	0.5875 ± 0.0542	0.5707 ± 0.0548	0.6229 ± 0.0598	0.6149 ± 0.0591	0.5945 ± 0.062	0.5981 ± 0.0580
Unaugmented	0.4873 ± 0.0385	0.483 ± 0.0349	0.484 ± 0.04	0.4862 ± 0.0406	0.4932 ± 0.0191	0.4867 ± 0.0346

Table 12: Average balanced accuracy on out-of-domain CPTAC test data of different classifier models in predicting the MSI status from embeddings retrieved from a VAE and an overall average of these models for different augmentation methods. A portion of the real training dataset (10%) was used for augmentation and each class was augmented to size 500.

Sampling Methods	EBM	KNN	Logistic	RF	SVM-RBF	Average
Mod. Gamma-Poisson	0.5808 ± 0.0826	0.6444 ± 0.0957	0.6823 ± 0.1153	0.6208 ± 0.0896	0.6458 ± 0.1118	0.6348 ± 0.0990
Inter-Crossover (Ours)	0.6601 ± 0.1057	0.7332 ± 0.1307	0.7087 ± 0.1329	0.6897 ± 0.104	0.728 ± 0.1256	0.7039 ± 0.1198
Intra-Crossover (Ours)	0.6394 ± 0.0859	0.7097 ± 0.1083	0.7333 ± 0.122	0.7016 ± 0.1163	0.7592 ± 0.1111	0.7086 ± 0.1087
Mod. Poisson	0.6345 ± 0.0792	0.6534 ± 0.1046	0.6996 ± 0.107	0.668 ± 0.0658	0.6463 ± 0.097	0.6604 ± 0.0907
Replacement	0.6495 ± 0.0701	0.637 ± 0.1028	0.7616 ± 0.1182	0.7021 ± 0.1014	0.6915 ± 0.1133	0.6883 ± 0.1012
SMOTE	0.6152 ± 0.0916	0.5956 ± 0.1044	0.6992 ± 0.1062	0.6892 ± 0.0947	0.6456 ± 0.089	0.6489 ± 0.0972
Unaugmented	0.5002 ± 0.0274	0.5144 ± 0.0411	0.4903 ± 0.0425	0.5078 ± 0.0374	0.4989 ± 0.0265	0.5023 ± 0.0350

Table 13: Average balanced accuracy on in-domain TCGA test data of different classifier models in predicting the CIMP status from embeddings retrieved from a VAE and an overall average of these models for different augmentation methods. A portion of the real training dataset (10%) was used for augmentation and each class was augmented to size 500.

Sampling Methods	EBM	KNN	Logistic	RF	SVM-RBF	Average
Mod. Gamma-Poisson	0.4329 ± 0.0757	0.5189 ± 0.0562	0.5275 ± 0.0344	0.5153 ± 0.0572	0.4867 ± 0.062	0.4963 ± 0.0571
Inter-Crossover (Ours)	0.4606 ± 0.0789	0.5633 ± 0.0572	0.5565 ± 0.0489	0.5183 ± 0.0597	0.5156 ± 0.0671	0.5229 ± 0.0624
Intra-Crossover (Ours)	0.4538 ± 0.078	0.5358 ± 0.0571	0.5451 ± 0.0476	0.5085 ± 0.0688	0.5076 ± 0.0624	0.5102 ± 0.0628
Mod. Poisson	0.4418 ± 0.0718	0.5024 ± 0.0705	0.5302 ± 0.0538	0.531 ± 0.0763	0.4818 ± 0.0797	0.4974 ± 0.0704
Replacement	0.4502 ± 0.0639	0.514 ± 0.0661	0.542 ± 0.0526	0.5288 ± 0.0801	0.5127 ± 0.0645	0.5095 ± 0.0654
SMOTE	0.4595 ± 0.0902	0.4968 ± 0.0615	0.5441 ± 0.0559	0.5308 ± 0.0699	0.5018 ± 0.0779	0.5066 ± 0.0711
Unaugmented	0.3452 ± 0.0392	0.3429 ± 0.0595	0.3363 ± 0.032	0.3411 ± 0.0646	0.3386 ± 0.0183	0.3408 ± 0.0427