GAN-based Data Augmentation for prediction improvement using gene expression data in cancer

Francisco J. Moreno-Barea$^{1*[0000-0002-3887-9095]}$, José M. Jerez$^{1[0000-0002-7858-2966]}$, and Leonardo Franco$^{1[0000-0003-0012-5914]}$

Departamento de Lenguajes y Ciencias de la Computación, Escuela Técnica Superior de Ingeniería Informática, Universidad de Málaga, Málaga, Spain.
{fjmoreno, jja, lfranco}@lcc.uma.es

Abstract. Within the area of bioinformatics, Deep Learning (DL) models have shown exceptional results in applications in which histological images, scans and tomographies are used. However, when gene expression data is under analysis, the performance is often limited, further hampered by the complexity of these models that require several instances, in the order of thousands, to provide good results. Due to the difficulty and the costs involved in the collection of medical data, the application of Data Augmentation (DA) techniques to alleviate the lack of samples is a topic of great relevance. State-of-the-art models based on Conditional Generative Adversarial Networks (CGAN) and some introduced modifications are used in this work to investigate the effect of DA for prediction of the vital status of patients from RNA-Seq gene expression data. Experimental results on several real-world data sets demonstrate the effectiveness and efficiency of the proposed models. The application of DA methods significantly increase prediction accuracy, leading by 12% with respect to benchmark data sets and 3.15% with respect to data processed with feature selection. Results based on CGAN models outperform in most cases, alternative methods like the SMOTE or noise injection techniques.

Keywords: Data augmentation · Gene expression · Bioinformatics · Deep Learning · CGAN.

1 Introduction

Deep learning (DL) models have become the state-of-the-art prediction algorithms in several application tasks, translating into billions of dollars invested by industries towards its application. With the advancement of deep network architectures, the access to large databases and the use of powerful computing systems, DL models have made incredible progress in a large variety of problems. DL models have a more complex structure compared to traditional machine learning methods, as they include thousands of parameters and dozens of layers that must be adjusted during the training process, and because of this, its application requires the use of large data sets with thousands of instances in order achieve a better performance than traditional machine learning techniques (shallow ANNs, SVMs, RF, etc.) [22, 7].

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08757-8_3

In particular, in the area of bioinformatics large and readily available data sets are scarce. Medical records are sensitive data with associated privacy problems and a difficulty to obtain patient consent for massive dissemination. Further, gene expression data are significantly more difficult to obtain, they present a greater dispersion and are prone to suffer from the curse of dimensionality, as microarray data contains a greater number of features compared to the number of samples usually available. For these reasons, the application of data augmentation (DA) methods has become one of the relevant topics in the area, allowing for the addition of new synthetic generated samples. A revision of recent state-of-the-art works in the field related to DL models applied to genomic data sets showed that some advantages can be observed using these models [4, 26] but we have not found works applying DA as it is used in the present work with the aim of improving prediction capabilities.

Like it happens with DL models, DA best results are found in computer vision and image processing areas, where data possesses structure. Specifically, DA models have shown impressive results in generating synthetic realistic images, based on a framework called Generative Adversary Networks (GAN) [8, 19]. Essentially, a GAN model network generates new samples from a distribution learned from the original data set, and for this purpose, the GAN produces a confrontation between two competing neural networks that learn from each other. Apart from the achievements of GAN models obtained in image vision, they have proven to be useful also for the DA task with images [9, 6, 27]. Applying DA to non-image data sets is far more challenging. Experts in an specific domain can be asked to assess the quality of a generated image and to distinguish a synthetic from real samples. However, this type of human expert based evaluation is not feasible when applied to non image-sets, even less if we take into account gene expression data. Most common methods for applying DA to non-structured data are the SMOTE technique (synthetic minority oversampling technique) [3] designed to deal with imbalanced data sets, and the noise injection methods as a way to prevent overfitting and improve prediction accuracy [20, 29, 18, 17]. Nonetheless, in recent times GAN models have become one of the reference DA methods also with other types of structured data such as time series or signals [11, 23], with data sets without any type of spatial or temporal structure [5, 21, 16], and also in biomedical problems [12, 14, 1].

Taking into account all the aspects mentioned above, this work has several objectives. Current research attempts to add knowledge to the existing scientific literature related to the application of DA with GAN models in biomedical problems, and more specifically with gene expression data. On the other hand we analyse modifications to state-of-the-art DA methods in order to obtain an increase in the precision of the cancer prognosis prediction problem compared to the traditional SMOTE and noise injection methods, which will allow the efficient application of techniques of Deep learning-based DA to small and non-structured data sets across multiple domains. Finally, we want to verify the methods ability to replicate the gene expression data with the Fréchet Inception Distance (FID), and be able to provide support for the prediction results.

2 Methodology

We include in this section the Data Augmentation (DA) methods and models applied to a cancer prognosis problem with different gene expression data sets.

2.1 Noise injection method

To perform DA with image sets there are some methods whose execution and approach is simple, such as resampling, flipping, cropping, shifting, or noise injection. To perform DA with non-image sets, some of these methods can also be used, such as resampling, based on repeating random instances of the data, or noise injection, based simply on modifying instances with degrees of noise. Although the noise injection may have a simple approach, the application of a procedure based on this method can be modified to obtain effective results [17].

The noise injection method designed randomly selects training samples and modifies a maximum of 25% of the features. The noise is generated from a random normal distribution with a standard deviation of 0.2 and is added to the original value of the feature, being subsequently controlled so as not to exceed the range of $[0; 1]$. A standard deviation value of 0.2 is enough to create samples that does not stray too far from the real space of instances.

2.2 SMOTE techniques

Apart from the addition of noise to perform DA with non image data sets, in the literature we can find some applications of SMOTE techniques (synthetic minority oversampling technique) [3] designed to generate synthetic data in data sets that present imbalanced classes. This oversampling technique uses a k-nearest neighbour algorithm, instead of random sampling with replacement. SMOTE performs a random interpolation of the instance of the selected minority class and its nearest neighbours, in order to balance the data set and operating in the feature space. The interpolation calculates the difference between the instance and each of the selected neighbours, multiplies the difference for each feature by a random normalisation and adds this value to the original feature of the sample. This process creates new instances of the minority class that are located within this space between the sample and its neighbours.

However, this technique has certain drawbacks due to random interpolation. One of the most notable disadvantages is the possible generation of samples that do not respect the geometry present in the data set. The generated samples can occupy positions in the feature space that belong to the majority class data. Other significant drawback is that SMOTE does not allow to control the amount of synthetic samples generated, only those necessary to balance the data set.

2.3 Conditional Generative Adversarial Networks

The standard GAN model [8] has a general structure composed by two neural networks, called the *generator* and the *discriminator*, that are trained simultaneously resulting in a confrontation process. In this way, the discriminator network

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08757-8_3

(D) tries to distinguish whether a sample comes from the real distribution or is a synthetic sample, i.e., for the input sample x, the discriminator estimates the probability that it belongs to the real distribution or not. The generator network (G) gets as output a synthetic sample from a noisy random distribution z. The purpose of the generator is create new synthetic samples with features that approximate those present in the real samples, so that the discriminator network will not be able to distinguish these synthetic samples as samples not coming from the real distribution. Therefore, the generator process is opposite to that of the discriminator, giving rise to a competitive environment.

Specifically, the model considered was the Conditional GAN (CGAN) [15], a variant of the standard GAN model. In CGAN, the information concerning to a condition y, the sample label or other data information is taken into account in the network. In this way, the latent space z and the condition y are passed as input to the generator network. This condition can be created randomly when training the model and it can be controlled when generating synthetic samples. The condition is also added to the input of the discriminator network, being the same that has been used to create a synthetic sample by the generator or the label assigned to the real sample.

$$\min_G \max_D \; \mathbb{E}_{x \sim p_{data}(x)}[\log D(x|y)] + \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z|y)))] \tag{1}$$

The objective cost function (Eq. 1) of the CGAN model presents the behaviours identified with the competitive process: one related to better recognise samples that belong to the real distribution and another related to better recognise samples created by the generator. In this way, the ability of the model to perceive whether the samples are real or fake is expressed in Eq. 2, and the error identified with the recognition of fake samples is expressed by Eq. 3.

$$\max_D \; \mathbb{E}_{x \sim p_{data}(x)}[\log D(x|y)] + \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z|y)))] \tag{2}$$

$$\min_G \mathbb{E}_{z \sim p_z(z)}[\log(D(G(z|y)))] \tag{3}$$

2.4 CGAN modified Generative Process

Considering the DA process to deal with supervised benchmark problems, we implemented modifications to the standard CGAN generative process giving rise to the ModCGAN model. This modified model was developed in a previous work [16]. The most significant difference from the generative process performed with ModCGAN compared to CGAN is the use of an external classifier called “generative classifier”. This generative classifier is used to label the synthetic samples created by the generator and discard them if they do not present enough quality. The whole generative process is shown in Fig. 1. The generative classifier is trained with the real samples from the training set, also adding noisy samples from two different methods: a uniform random distribution and gaussian noise injection. The use of these different noise sources teaches the classifier to distinguish real and fake samples, from pure noise (uniform random distribution) and samples with similar aspects to the real distribution (gaussian noise injection).

In the ModCGAN generative process, the generator creates a synthetic sample from a noisy random distribution and a label, since its generative base is the same as CGAN. However, instead of using the discriminant network, ModCGAN uses the generative classifier to estimate the label for the synthetic sample. If the 'noise' label is estimated, sample is considered fake and is discarded. On the other hand, if the estimated label is different from 'noise', it means that classifier has predicted a label from the real ones, so the sample is saved with the predicted label. Applying this modified process, the synthetic sample may be assigned a different label than the one used by the generator at the generative process. Furthermore, it is possible that samples that the discriminator network could consider fake are saved or, conversely, not save samples that the discriminator could detect as real but that the generative classifier predicts as 'noise'.

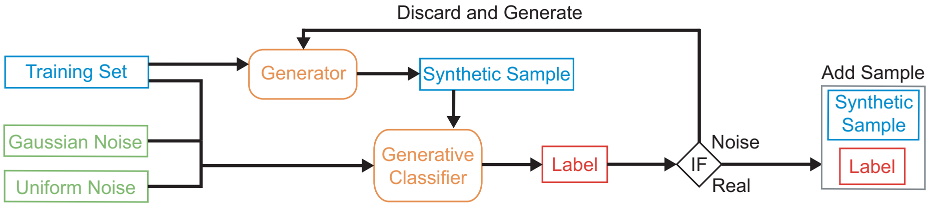

Fig. 1. Generative DA process for the creation of synthetic data in the ModCGAN model. When the generative model and the generative classifier are trained, the synthetic sample is generated and the appropriate label is predicted. If the label represents noise, the sample is discarded and another sample is generated, else the sample is save.

2.5 Modifications for treating unbalanced data set distribution

A possible problem that arises from generating synthetic samples with GAN models is the so-called collapse of the model. This problem arises when the generator creates the same synthetic samples repeatedly. Also when it creates synthetic samples from only one of the classes, usually from the majority class. Two modifications, 'Multiclass' and 'Balanced Multiclass' [16], were applied to avoid collapse of the model causing problems in the DA application. The modification was indicated in the models with the suffixes '_M' and '_BM' respectively.

The GAN-based models with these modifications use two independent models. Each independent model is trained with a set that presents all the samples that belong to one of the classes with a random selection of samples from the other class, trying to reach a ratio of 20%. The purpose is that each generator is able to focus on one class of the problem, always taking into account its differences with the samples of the other class. These generators are used in an ensemble methods process, each generating a specific proportion of samples that are joined to produce the final synthetic data set.

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08757-8_3

The specific number of samples generated by each generator depends on the desired number of samples, the proportion of each class in the original data set, and the implemented modification applied. The generative process with 'Multiclass' tries to keep the original classes proportion, while with 'Balanced Multiclass' it generates more samples for the minority class. It should be mentioned that the generative process followed in both modifications remains as the original GAN-based model, so if a sample is classified as fake or noise, the same generator model that created this sample generates a new one. These modifications are not intended to force the model to generate only synthetic samples that belong to one of the classes, but rather to better adapt the distribution of the samples to avoid collapse when the model trains with all the data.

2.6 Implementations of the models

The CGAN and ModCGAN models, and the balance control modifications, present the same standard implementation with the exception of the specific implementation of the ModCGAN generative classifier. Generator network presents 4 hidden layers with Rectified Linear Unit (ReLU) [28] as activation function (de facto state-of-the-art activation functions in DL). Discriminator network also presents 4 hidden layers but with Leaky ReLU [13] activation function, since it provides more stability than ReLU in classification tasks. Both networks used batch normalisation as regularisation technique and Adam algorithm as optimisation algorithm with adaptive learning rate.

On the other hand, the classifier model used for the classification experiments and the generative classifier for synthetic process was a deep feedforward neural network. It presents 3 hidden layers with Leaky ReLU activation function and batch normalisation and dropout [24] at each hidden layer. The sigmoid activation function was used in the output neuron to classify patterns. Adam algorithm was also used as optimisation algorithm. The dropout rate applied was 0.1, 0.5 and 0.3 in the hidden layers of the generative classifier, and 0.3, 0.6 and 0.4 in the classifier used in test prediction experiments. The L2 norm was used in combination with dropout and batch normalisation to avoid overfitting.

2.7 Benchmark data sets

The benchmark data sets used for the experiments are freely available at The Cancer Genome Atlas (TCGA) website, provided by International Cancer Genome Consortium (ICGC). The data sets correspond to patients linked to 18 different cancer types: bladder carcinoma (blca), breast carcinoma (brca), colon adenocarcinoma (coad), glioblastoma multiforme (gbm), head and neck squamous cell carcinoma (hnsc), pan-kidney cohort (kipan), kidney renal cell carcinoma (kirc), brain lower grade glioma (lgg), liver hepatocellular carcinoma (lihc), lung adenocarcinoma (luad), lung squamous cell carcinoma (lusc), ovarian carcinoma (ov), prostate adenocarcinoma (prad), skin cutaneous melanoma (skcm), stomach adenocarcinoma (stad), stomach and esophageal carcinoma (stes), thyroid adenocarcinoma (thca) and uterine corpus endometrioid carcinoma (ucec).

The instances of these data sets represents patients affected of cancer, and for each patient it contains a row of 20531 variables than correspond to the expression level of a certain gene, so the data sets are RNA-Seq gene expression profiles after applying pre-processing procedures for batch correction and RSEM normalisation. A logarithmic (log2) transformation of the expression levels in the data was carried out, to approximate them to a normal distribution for its use with the predictive models. Additionally, a feature selection process was applied using the LASSO model [25] and the Gini importance from Random Forest method [2], reducing the number of genes. In order to perform a prediction analysis, vital status information for each patient has been collected, which is also freely available in TCGA. The vital status therefore supposes the label present in the data and the objective to be predicted.

$$\text{Balance} = \frac{H}{\log k} = \frac{-\sum_{i=1}^{k} \frac{c_i}{n} \log \frac{c_i}{n}}{\log k} \tag{4}$$

Table 1 shows some characteristics of the benchmark data sets, the columns show the name of the benchmark data set, the number of features after feature selection (Feat.) and instances (Inst.), the proportion of classes (Bal.), and the most significant gen according to the feature selection (Sig-Gen). Instead of showing the percentage of instances that belong to each class, we show a measure of balance (Eq. 4) based on the Shannon entropy (H). This measure is calculated given the number of instances n in the data set, the number of classes k, and the size of each class c_i. If the value of Balance is 1, the set is completely balanced, and if the value is 0, the set is completely unbalanced.

Table 1. Characteristics of the eighteen gene expression data sets studied.

Data	Feat.	Inst.	Bal.	Sig-Gen
blca	114	427	0.99	SPG7
brca	74	1212	0.64	ZNF331
coad	12	191	0.71	ALPK3
gbm	19	171	0.73	ABCB8
hnsc	10	566	0.99	SLC25A43
kipan	102	1020	0.83	BANP
kirc	88	606	0.92	DPAGT1
lgg	57	242	0.98	CCNI
lihc	11	423	0.96	EIF5B
luad	13	344	0.96	OR2T335
lusc	18	552	0.99	PYGB
ov	33	307	0.97	PERP
prad	27	550	0.13	SNORA16A
skcm	21	473	1.00	INSR
stad	3	450	0.96	LPPR2
stes	47	646	0.97	PRTG
thca	21	568	0.22	CXCL5
ucec	31	201	0.67	PEX11A

3 Experiments and Results

In order to keep complete independence between data generation, classifier model training, and prediction accuracy evaluation, we performed a division of the data set into training, validation and test sets. The synthetic data generation does not include any samples from the test set, which is kept separate for honest external performance testing. A 10-fold cross validation procedure is implemented in the prediction experiments and the training folds are augmented with synthetic samples. The result of the classification process is the average of the accuracy results obtained for the 10-folds. This process is further repeated with 10 different seeds to reduce possible random effects.

Table 2 shows the results obtained for the 18 data sets studied and described previously in Table 1. First column shows the test accuracy obtained with the original 'raw' data set, second column ('FS') shows the test accuracy when feature selection pre-processing is applied but not including any Data Augmentation process. Next group of columns show results obtained when DA is applied on the data sets after the feature selection process, showing the DA method used, the test accuracy obtained and the percentage of augmentation applied to the training data set. The model with the highest accuracy evaluated on the validation set is the indicated one. Last columns in the Table 2 show the relative difference (Eq. 5) obtained for each of the three DA methods applied: CGAN-based models, SMOTE and noise injection. The reference results for calculating the RD are the results obtained with the feature selection pre-processing. Last column, $\widehat{RD}$, shows the maximum value for the relative difference over the results obtained previously (indicated with bold font).

$$\text{RD} = \frac{(\text{Acc\_Aug} - \text{Acc\_Ref})}{\text{Acc\_Ref}} \times 100 \tag{5}$$

Table 2. Test accuracies obtained with the original data set (col. 2) and when a feature selection method is applied (col. 3). Cols. 4-6 show the test accuracy for the best case of the three implemented DA methods and the corresponding percentage of generated samples. Cols. 7-10 shows Test RD for the three used methods and last column ($\widehat{RD}$) the best RD obtained (see text for details).

			Data Augmentation			RD			
Data	Original	FS	Acc.	Model	Perc.	CGAN	SMOTE	NOISE	$\widehat{RD}$
gbm	0.6353	0.7794	0.8185	CGAN	200	**5.02**	-0.45	-1.21	5.02
coad	0.7015	0.7421	0.7529	CGAN	50	**1.45**	-5.82	-0.50	1.45
ucec	0.7250	0.7955	0.8405	NOISE	200	1.98	1.29	**5.66**	5.66
lgg	0.6802	0.8313	0.8413	CGAN_M	50	**1.20**	-1.03	0.70	1.20
ov	0.5192	0.7370	0.7921	ModCGAN_M	200	**7.47**	2.62	-6.56	7.47
luad	0.6071	0.6368	0.6694	CGAN_M	100	**5.13**	0.35	-1.48	5.13
lihc	0.6248	0.6690	0.6918	NOISE	200	1.19	1.64	**3.40**	3.40
blca	0.5566	0.7412	0.7636	CGAN_M	50	**3.03**	0.24	-0.76	3.03
stad	0.6093	0.6623	0.6708	SMOTE	None	0.79	**1.28**	-0.91	1.28
skcm	0.5213	0.6872	0.7377	ModCGAN_BM	200	**7.34**	-0.37	-2.94	7.34
prad	0.8918	0.9772	0.9810	SMOTE	None	0.31	**0.39**	0.04	0.39
lusc	0.5485	0.5624	0.5857	CGAN_BM	200	**4.15**	0.15	-4.98	4.15
hnsc	0.5919	0.7000	0.7388	ModCGAN_BM	200	**5.54**	1.50	-5.90	5.54
thca	0.9106	0.9640	0.9655	NOISE	200	0.05	-0.03	**0.16**	0.16
kirc	0.7239	0.7836	0.8041	ModCGAN_BM	100	**2.62**	0.01	2.04	2.62
stes	0.5747	0.5895	0.6033	CGAN_M	50	**2.35**	-0.43	0.03	2.35
kipan	0.7402	0.8265	0.8278	ModCGAN	200	**0.17**	-2.85	-1.31	0.17
brca	0.8072	0.8531	0.8567	ModCGAN	100	**0.42**	-0.90	0.09	0.42
Mean	0.6667	0.7521	0.7745			**2.79**	-0.13	-0.80	3.15

The results on Table 2 show an average improvement of 3.15 % in $\widehat{RD}$ and 2% in test prediction accuracy using DA methods compared to the case when only feature selection is implemented. Using the FS pre-processing and DA methods a substantial improvement of approximately 11% is achieved over the original raw benchmark data sets, noting that the feature selected data set already permit to achieve almost a 9% increase in accuracy over the original data set, and thus obtaining a further increase with DA techniques is a relevant achievement.

The results indicate that for 13 out of 18 data sets best accuracy results are obtained through DA based on CGAN models, also obtaining in this case the best average results with a RD value of 2.79 % . The noise injection method is the best one for 3 data sets and finally SMOTE leads in two cases; however both methods lead to negative RD values. In addition, 12 of 18 methods generate a percentage of samples greater than or equal to 100, so the models at least double the number of samples present in the training set. Regarding the efficacy of the different CGAN-based models (i.e. this analysis does not take into account noise injection or SMOTE methods), ModCGAN is the best option for 10 data sets while CGAN is the preferred method for the remaining 8.

Figure 2 shows the relationship between the test accuracy obtained for the data sets with FS and DA application and the obtained using the original raw benchmark data sets. The results show that the improvement in accuracy obtained from DA with respect to FS and the original results tends to be greater for those sets whose original accuracy is lower.

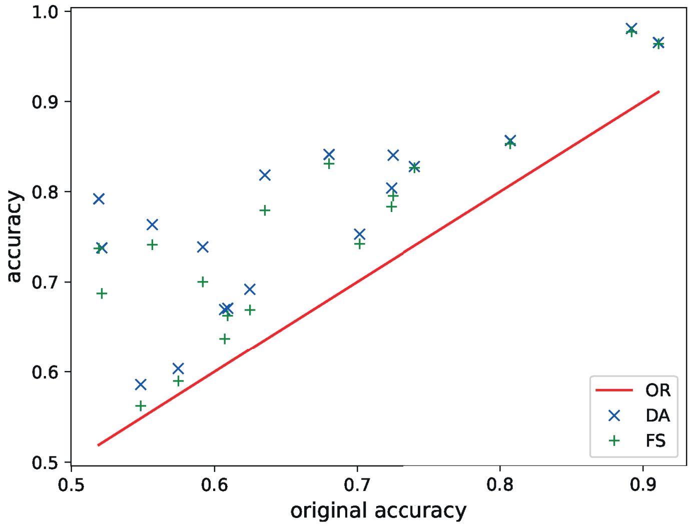

Fig. 2. Relationship between accuracy obtained with the processed data sets (DA and FS) and the original ones. Crosses represent the results obtained with augmented data sets and cross-hairs those obtained when only feature selection is applied. The continuous line represents the identity function.

DOI: 10.1007/978-3-031-08757-8_3

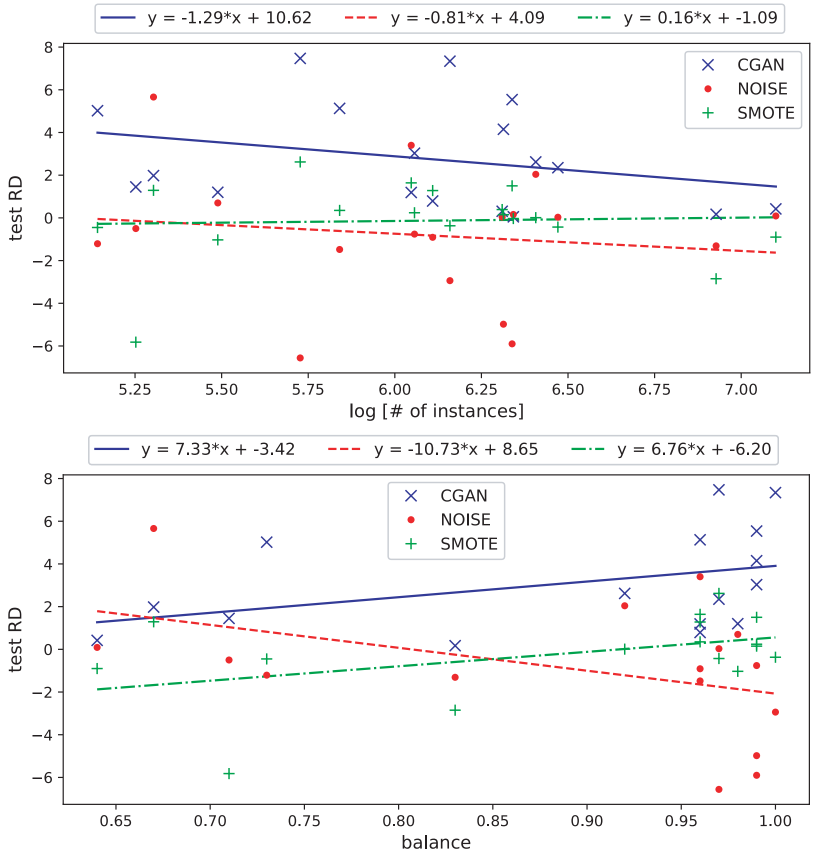

Fig. 3. Relative prediction accuracy difference (RD) *vs* the logarithm of the number of instances (top) and *vs* the balance (bottom). The lines are a linear regression adjusted to the data. Crosses and continuous line represent the results obtained with the CGAN model, dots and dashed line those obtained with the NOISE method, and cross-hairs and dashed-dotted line those obtained with SMOTE.

To analyse the influence of the data sets size on the precision obtained, we graph in Fig. 3 (top) the test relative difference RD obtained with each DA method versus the number of instances in a logarithmic scale on the x-axis. A linear regression model was fit to results for each type of DA method, obtaining for the CGAN-based models a correlation coefficient of -0.29, which indicates a moderate negative correlation between the number of instances and the prediction accuracy gain. Similar with the linear regression model fitted for the noise injection results, with a correlation coefficient of -0.14. The results obtained with SMOTE show that the relationship between the number of instances and the prediction accuracy gain remains stable, with a correlation coefficient of 0.04.

In a similar way, Fig. 3 (bottom) shows the test relative difference RD obtained with each DA method versus the balance degree. Figure results does not include data sets prad and thca due to their low balance value with respect to the rest (0.13 and 0.22 respectively), being outliers and preventing a good visualisation of the results. The correlation coefficient obtained for the CGAN-based models is 0.4, indicating a positive correlation between balance and prediction accuracy gain. The linear regression model fitted with SMOTE results is similar, with a correlation coefficient of 0.44. On the contrary, the results with the addition of noise method indicate a negative correlation between balance and prediction accuracy gain, with a correlation coefficient of -0.43.

In order to analyse how the DA methods used in the experimentation were able to replicate the information present in the gene expression data set, the Fréchet Inception Distance (FID) [10] is computed. FID is a metric used to measure the quality of the images generated by GAN models, but FID is also applicable to any data generation application. Equation 6 shows the FID calculation to compare the distribution r with the distribution g from mean values of the real (μ_r) and the generated (μ_g) vectors, the trace of the matrix (Tr) and the covariance matrix of the vectors (Σ_r, Σ_g).

$$\mathrm{FID} = \|\mu_r - \mu_g\|^2 + \mathrm{Tr}(\Sigma_r + \Sigma_g - 2(\Sigma_r \Sigma_g)^{1/2}) \tag{6}$$

Table 3. Mean Fréchet Inception Distance ($\widehat{FID}$) obtained with each data augmentation model with respect to the real Train and Test distributions for each data set. Lower values indicate more similarity between synthetic and real samples.

	SMOTE		NOISE		CGAN		ModCGAN	
Data	Train	Test	Train	Test	Train	Test	Train	Test
gbm	0.150	0.893	0.179	0.602	0.407	0.769	0.206	0.644
coad	0.076	0.469	0.072	0.334	0.451	0.771	0.147	0.377
ucec	0.287	1.699	0.299	1.232	1.305	1.790	0.741	1.440
lgg	1.015	1.615	0.351	1.486	2.082	2.686	1.359	2.224
ov	0.264	0.490	0.103	0.434	0.745	1.099	0.256	0.554
luad	0.040	0.133	0.028	0.101	0.283	0.314	0.046	0.103
lihc	0.030	0.046	0.018	0.040	0.089	0.124	0.021	0.046
blca	1.441	2.037	0.541	2.060	3.170	3.820	2.057	2.471
stad	0.002	0.005	0.003	0.007	0.112	0.099	0.042	0.039
skcm	0.175	0.266	0.041	0.134	0.277	0.429	0.107	0.196
prad	0.269	0.000	0.047	0.094	1.020	1.795	1.142	2.687
lusc	0.086	0.135	0.029	0.093	0.280	0.326	0.051	0.105
hnsc	0.028	0.039	0.011	0.028	0.111	0.146	0.015	0.033
thca	0.223	1.083	0.222	0.623	0.801	1.265	0.411	0.804
kirc	0.378	0.949	0.341	1.127	3.413	3.657	1.359	1.708
stes	0.171	0.390	0.143	0.394	1.440	1.640	0.576	0.771
kipan	0.212	0.794	0.319	0.971	3.448	3.624	1.818	2.096
brca	0.128	0.700	0.206	0.682	2.064	2.336	1.052	1.291
Mean	0.276	0.652	0.164	0.580	1.194	1.483	0.634	0.977

Mean FID values ($\widehat{FID}$) are obtained from the comparison of the synthetic class 0 distribution with the real class 0 distribution and the synthetic class 1 distribution with the real class 1 distribution. The $\widehat{FID}$ results for SMOTE only refer to the minority class comparison. Table 3 shows the $\widehat{FID}$ values obtained in the comparison with the real Train and Test distributions with the synthetic set generated with the different DA methods used, making a comparison between CGAN and ModCGAN.

The results reported in the Table 3 reveal that the addition of noise generates samples with great similarity to the original sample distribution. Samples generated by this method obtain the lowest $\widehat{FID}$ value, 0.164 on average measured on training samples. For the case of applying the SMOTE method, the analysis reveals a level of similarity slightly greater than the noise based one, with low $\widehat{FID}$ values (0.28 and 0.65 with respect to train and test). Regarding GAN-based methods, these add more variability to the augmented data sets, as samples generated with CGAN have less similarity for train and test data sets, reaching the highest average values of $\widehat{FID}$ (1.19 and 1.48 with respect to training and testing). On the other hand, the samples generated with ModCGAN model present lower values of $\widehat{FID}$, which indicates a greater similarity with the real samples. The $\widehat{FID}$ values are almost half of those obtained with CGAN (0.63 and 0.98).

4 Conclusion

In this work, we proposed the application of different state-of-the-art techniques for Data Augmentation (DA), with the aim of improving the prediction accuracy in patient prognosis analysis that can be obtained when data sets of RNA-Seq gene expression profiles are studied in different types of cancer. The results indicate that the application of DA methods can lead to an increase in prediction accuracy of approximately 3% (all the tested methods are evaluated, choosing the best one according to the validation error). This improvement has been achieved with respect to the data sets after applying a feature selection technique, as the improvement in prediction accuracy over the original raw data is approximately 11%. We observed also that conditional GAN models can greatly improve the generalisation results as a 2.79% increase was obtained, while alternative models like SMOTE and noise injection lead to negative results. Additionally, the quality of the generated samples was analysed to explain the performance achieved by each DA methods, and for this purpose the Fréchet Inception Distance (FID) was measured. From this analysis, we concluded that the noise addition method generates more similar samples, while CGAN-based models offers more variability. In the light of these results, we can draw the conclusion that greater variability in the augmented sets increases the potential of the prediction models to correctly classify test samples (never presented before to the classification model) that may not be similar to training ones.

In conclusion, DA techniques constitute a suitable approach to increase the prediction performance in patient prognosis analysis with data sets of RNA-Seq gene expression profiles. DA techniques based on CGAN models are capable of

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08757-8_3

generating good quality synthetic data that lead on average to a 3% relative prediction increase. In relation to this, several future studies are planned, extending the application of DA methods to other gene expression data sets and bioinformatics tasks.

Acknowledgements. The authors acknowledge the support from MICINN (Spain) through grant TIN2017-88728-C2-1-R and PID2020-116898RB-I00, from Universidad de Málaga y Junta de Andalucía through grant UMA20-FEDERJA-045, and from Instituto de Investigación Biomédica de Málaga – IBIMA (all including FEDER funds). The results published here are based upon data generated by the TCGA Research Network: https://www.cancer.gov/tcga.

References

1. Barile, B., Marzullo, A., Stamile, C., Durand-Dubief, F., Sappey-Marinier, D.: Data augmentation using generative adversarial neural networks on brain structural connectivity in multiple sclerosis. Computer Methods and Programs in Biomedicine **206**, 106113 (2021). https://doi.org/10.1016/j.cmpb.2021.106113
2. Breiman, L.: Random forests. Machine learning **45**(1), 5–32 (2001). https://doi.org/10.1023/a:1010933404324
3. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: synthetic minority over-sampling technique. Journal of artificial intelligence research **16**, 321–357 (2002). https://doi.org/10.1613/jair.953
4. Cheerla, A., Gevaert, O.: Deep learning with multimodal representation for pancancer prognosis prediction. Bioinformatics **35**(14), i446–i454 (2019). https://doi.org/10.1093/bioinformatics/btz342
5. Douzas, G., Bacao, F.: Effective data generation for imbalanced learning using conditional generative adversarial networks. Expert Systems with Applications **91**, 464–471 (1 2018). https://doi.org/10.1016/j.eswa.2017.09.030
6. Frid-Adar, M., Diamant, I., Klang, E., Amitai, M., Goldberger, J., Greenspan, H.: GAN-based synthetic medical image augmentation for increased CNN performance in liver lesion classification. Neurocomputing **321**, 321–331 (12 2018). https://doi.org/10.1016/j.neucom.2018.09.013
7. Goodfellow, I., Bengio, Y., Courville, A.: Deep learning. MIT press (2016)
8. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative Adversarial Nets. In: Advances in Neural Information Processing Systems. pp. 2672–2680 (2014)
9. Han, C., Hayashi, H., Rundo, L., Araki, R., Shimoda, W., Muramatsu, S., Furukawa, Y., Mauri, G., Nakayama, H.: GAN-based synthetic brain MR image generation. In: 2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018). pp. 734–738. IEEE (2018). https://doi.org/10.1109/isbi.2018.8363678
10. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. In: Advances in Neural Information Processing Systems. vol. 30 (2017)
11. Hsu, W.N., Zhang, Y., Glass, J.: Unsupervised domain adaptation for robust speech recognition via variational autoencoder-based data augmentation. In: 2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU). pp. 16–23. IEEE (12 2017). https://doi.org/10.1109/asru.2017.8268911

12. Liu, Y., Zhou, Y., Liu, X., Dong, F., Wang, C., Wang, Z.: Wasserstein GAN-Based Small-Sample Augmentation for New-Generation Artificial Intelligence: A Case Study of Cancer-Staging Data in Biology. Engineering **5**(1), 156–163 (2 2019). https://doi.org/10.1016/j.eng.2018.11.018
13. Maas, A.L., Hannun, A.Y., Ng, A.Y.: Rectifier Nonlinearities Improve Neural Network Acoustic Models. In: International Conference on Machine Learning. vol. 30, p. 3 (2013)
14. Marouf, M., Machart, P., Bansal, V., Kilian, C., Magruder, D.S., Krebs, C.F., Bonn, S.: Realistic in silico generation and augmentation of single-cell RNA-seq data using generative adversarial networks. Nature communications **11**(1), 1–12 (2020). https://doi.org/10.1038/s41467-019-14018-z
15. Mirza, M., Osindero, S.: Conditional generative adversarial nets (2014)
16. Moreno-Barea, F.J., Jerez, J.M., Franco, L.: Improving classification accuracy using data augmentation on small data sets. Expert Systems with Applications **161**, 113696 (2020). https://doi.org/10.1016/j.eswa.2020.113696
17. Moreno-Barea, F.J., Strazzera, F., Jerez, J.M., Urda, D., Franco, L.: Forward Noise Adjustment Scheme for Data Augmentation. In: IEEE Symposium Series on Computational Intelligence (IEEE SSCI 2018) (2018). https://doi.org/10.1109/ssci.2018.8628917
18. Piotrowski, A.P., Napiorkowski, J.J.: A comparison of methods to avoid overfitting in neural networks training in the case of catchment runoff modelling. Journal of Hydrology **476**, 97–111 (2013). https://doi.org/10.1016/j.jhydrol.2012.10.019
19. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks (2015)
20. Reed, R.D., Marks, R.J.: Neural Smithing: Supervised Learning in Feedforward Artificial Neural Networks. Cambridge, MA, USA: MIT press (1998)
21. dos Santos Tanaka, F.H.K., Aranha, C.: Data Augmentation Using GANs. Proceedings of Machine Learning Research XXX **1**, 16 (2019)
22. Schmidhuber, J.: Deep learning in neural networks: An overview. Neural networks **61**, 85–117 (2015). https://doi.org/10.1016/j.neunet.2014.09.003
23. Shao, S., Wang, P., Yan, R.: Generative adversarial networks for data augmentation in machine fault diagnosis. Computers in Industry **106**, 85–93 (4 2019). https://doi.org/10.1016/j.compindJ.2019.01.001
24. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. Journal of Machine Learning Research **15**(1), 1929–1958 (2014)
25. Tibshirani, R.: Regression shrinkage and selection via the Lasso. Journal of the Royal Statistical Society. Series B (Methodological) **58**(1), 267–288 (1996). https://doi.org/10.1111/j.2517-6161.1996.tb02080.x
26. Vale-Silva, L.A., Rohr, K.: Long-term cancer survival prediction using multimodal deep learning. Scientific Reports **11**(1), 1–12 (2021). https://doi.org/10.1038/s41598-021-92799-4
27. Waheed, A., Goyal, M., Gupta, D., Khanna, A., Al-Turjman, F., Pinheiro, P.R.: Covidgan: data augmentation using auxiliary classifier gan for improved covid-19 detection. Ieee Access **8**, 91916–91923 (2020). https://doi.org/10.1109/access.2020.2994762
28. Xu, B., Wang, N., Chen, T., Li, M.: Empirical evaluation of rectified activations in convolutional network (2015)
29. Zur, R.M., Jiang, Y., Pesce, L., Drukker, K.: Noise injection for training artificial neural networks: A comparison with weight decay and early stopping. Medical physics **36**(10), 4810–4818 (2009). https://doi.org/10.1118/1.3213517