Data Generation Using Gene Expression Generator

Zakarya Farou, Noureddine Mouhoub, Tomáš Horváth

Data Generation Using Gene Expression Generator

Zakarya Farou[1 [0000−0003−3996−2656]], Noureddine Mouhoub[2], and Tomáš Horváth[1,3 [0000−0002−9438−840X]]

[1] Department of Data Science and Engineering, Telekom Innovation Laboratories, Faculty of Informatics, ELTE - Eötvös Loránd University Pázmány Péter sétány 1/C, H-1117, Budapest, Hungary
{zakaryafarou,tomas.horvath}@inf.elte.hu
[2] Science and Technology Campus, Bordeaux Computer Science Laboratory - LaBRI, University of Bordeaux, 351 Cours de la Liberátion, F-33400, Talence, France
noureddine.mouhoub@labri.fr
[3] Institute of Computer Science, Pavol Jozef Šafárik University Jesenná 5, 040 01 Košice, Slovakia
http://t-labs.elte.hu/

Abstract. Generative adversarial networks (GANs) could be used efficiently for image and video generation when labeled training data is available in bulk. In general, building a good machine learning model requires a reasonable amount of labeled training data. However, there are areas such as the biomedical field where the creation of such a dataset is time-consuming and requires expert knowledge. Thus, the aim is to use data augmentation techniques as an alternative to data collection to improve data classification. This paper presents the use of a modified version of a GAN called Gene Expression Generator (GEG) to augment the available data samples. The proposed approach was used to generate synthetic data for binary biomedical datasets to train existing supervised machine learning approaches. Experimental results show that the use of GEG for data augmentation with a modified version of leave one out cross-validation (LOOCV) increases the performance of classification accuracy.

Keywords: Data generation · Generative adversarial networks · Gene expression data · Cancer classification

1 Introduction

Automated diagnosis of oncology diseases such as colon cancer is extremely complex and requires special attention. Gene expression profiling is widely adopted to learn and analyze the conditions of cells and their response to various conditions, which is useful in the pathogenesis of diseases. Gene expression microarrays can be used for diagnostic purposes as well as for insights into biology. *Gene Expression Data* (GED) is a high dimensional data with a high number of features that indicate gene levels, but with very few records. Usually a satisfying number

of measurements are available for each tumor sample, with each measurement belonging to a particular gene.

Obtaining accurate results while training a classifier becomes difficult due to the lack of available, varied and meaningful data. It is therefore recommended that additional samples be added to train the classifier more efficiently. Increasing the original data, i.e., generating additional samples from the existing ones, for the imbalanced class would avoid overfitting situations and improve the classification process.

1.1 Motivation

Building a satisfying Machine Learning (ML) model usually requires a large number of samples. In the biomedical field, the collection of such data is very expensive and time-consuming. Experts must store, examine and annotate the recorded data (which can be either an image, information or clinical tests) to obtain a clean, meaningful and useful dataset. Recently, Deep Neural Networks (DNNs) [9] has made many improvements in ML, especially when massive datasets are available. The popularity of DNNs has led to their use also in cases where only a small number of samples are available, and simpler ML techniques, requiring less computational effort, would be considerably better or even better than DNNs. However, in the case of a very limited number of data samples, it is difficult to train any type of ML model with reasonable performance. Thus, this work aims to highlight that even the performance of simple classification techniques can be improved by providing the appropriate augmentation technique.

In the classification of cell dysplasia (cancer data), the sensitivity of the data should be maintained. This means that when new instances are generated using data augmentation techniques, it is important to ensure that the generated data is close to the original data, not only in terms of values but also in terms of data semantics.

Generative Adversarial Networks (GANs) [6], established in recent years, have attracted the attention of researchers (Fig. 1). Several variants of GANs have been proposed to generate high-quality synthetic data, where they have been used for data augmentation in cases where traditional data augmentation methods do not yield good results. Thus, the feasibility of using GANs as a data augmentation technique to improve the performance of classifiers in GED classification is worth exploring. To our knowledge, the use of GANs in relation to cancer classification by gene expression has not yet been performed.

1.2 Contributions

As the collection of large amounts of medical data is costly and difficult to acquire due to certain privacy constraints, data generation is an alternative to data collection, especially when synthetic data can be generated from a small number of existing samples. To this end, this work provides several contributions.

First, a modified version of the GAN called *Gene Expression Generator* (GEG) is proposed. GEG learns the GED distribution and tries to synthesize new samples that are consistent with the original data. The GEG discriminator uses the Wasserstein distance to reflect the similarity between the original and synthetic data distributions, which allows for greater stability during training. GEG also uses data restriction, i.e., the discriminator is fed only with data belonging to a single class, which allows to avoid unnecessary steps leading to labeled generated data after GEG training. Although in this work GEG was used only for binary classification, it would also work for multi-class classification problems.

Next, the paper introduces a modified version of leave one out cross-validation (LOOCV) by not merging the synthetic data with the original data but using the generated instances only for the training of the classifiers. By using the generated data as an extension of the training samples, testing the model performance with the original data only and improving the results, more attention is paid to data sensitivity. The generated data helps the model to better understand the data without interfering with the original data.

2 Related work

The GAN, introduced by Goodfellow et al. [6], has quickly received increased interest and researchers have explored its capabilities in a wide variety of applications [28]. The most successful application of GAN is computer vision, including image translation [8], image super-resolution [10], image synthesis [29], video generation [26], face aging [2], 3D object generation [23] or detection of small objects [30]. Another area of application of GAN concerns natural language processing [5], speech processing [13], and text generation [27].

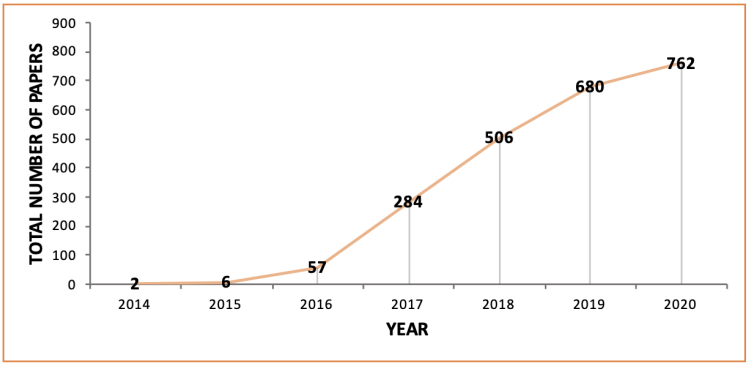

Fig. 1: Cumulative number of GAN-related paper publications/journals per year since its introduction in 2014

Several variants of the GAN have been proposed in recent years. A conditional version of the generative adversarial nets called CGAN [18] was introduced to extend the GAN by conditioning the generator and discriminator with additional information. The Deep Convolutional GAN (DCGAN) [20] has certain architectural constraints and is powerful while dealing with unsupervised learning problems. Another variant, called GRAN [7] was proposed to generate images with recurrent adversarial networks. MGAN [12] is a Markovian GAN, a technique for training generative neural networks for efficient texture synthesis. GAN-CLS [21] demonstrated the ability to generate plausible images of birds and flowers using simple text descriptions. An alternative to the GAN called VIGAN [22], an extended version of CycleGAN [31], handled missing data imputations from the MNIST.

A distributed adversarial network called DAN was proposed in [11] where the adversarial training relies on the entire sample as a unit and not on its sample points. Unlike researchers who usually use thousands of images, Marchesi et al. [16] investigated the possibility of applying GAN to produce high-quality megapixel images using a limited amount of data. Lu et al. [15] suggested a new approach called Bi-GAN which uses two generators dedicated solely to generating synthetic data from Gaussian noise, two evaluators and a discriminator for data classification. Later, Wang et al. [28] suggested working with a set of generators against a single discriminator. Finaly, Ian Goodfellow introduced a new robust and simple to train method called latent adversarial generator (LAG) [3], which generates high-resolution images using latent spaces.

The review of the GAN and its areas of application reveals that most researchers are focusing on image/video data generation, while there are still some unexplored areas such as biomedical field, which is a delicate, sensitive and costly area (in terms of data acquisition). Recently, there have been some attempts to use GAN for genetic data [17]. Due to the relatively small number of researchers working on the GAN for genetic data, this work aims to examine the GAN alongside the GED.

3 Proposed gene expression generator

Generative adversarial networks (GANs) are deep neural networks based on game theory [6]. They are considered as intelligent and creative machines. Fig. 2 shows that a GAN has two principal components: a generator G and a discriminator D (red boxes); G and D are neural networks. The output of G i.e. x' where $x' = G(z)$ is directly linked to the input of D next to the original data x. G produces synthetic instances x' starting from random Gaussian noise z such that x' and x are consistent. D checks the closeness of the synthetic data to the original data and returns a probability between 0 and 1 for each instance created. In the basic GAN, a cross-entropy loss is used to estimate the error between predicted/actual label of D output and the actual labels.

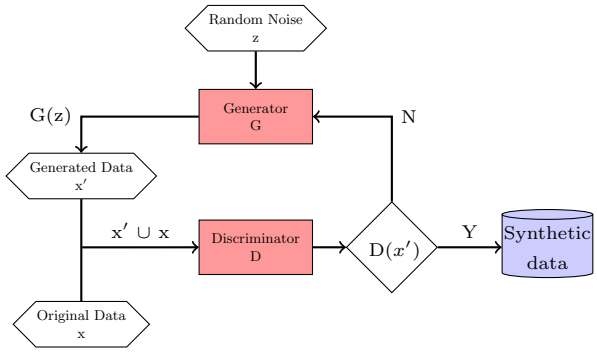

Fig. 2: Process of creating synthetic data using GAN

3.1 GEG architecture

The gene expression generator, i.e. GEG, is a variant of GAN that attempts to reproduce the probability distribution of gene expression. GEG uses original samples from a single class as input to generate more artificial instances referring to the same class label. The Wasserstein distance WD is used as a loss function to reflect the similarity between the distribution of the original and synthetic data generated by GEG. When a discriminator uses the Wasserstein loss function, it does not try to classify the instances by giving them a class label (i.e. synthetic or real instance), but attaches a number to each instance so that high values are used for the original data and low values for the synthetic data. However, in simple GANs, the discriminator gives a probability p for each instance, where $p \in [0..1]$ and, based on a threshold (e.g. 0.5), it can classify the instances either as artificial (negative) or original (positive) instances.

A GAN can use a unique loss function θ for both G and D. Therefore, one of them (G or D) should use $-\theta$ (the same loss function differing only by the sign). In this case, GEG uses different loss functions. It uses a generator loss $-\mathbf{D}(\mathbf{x}')$ where G tries to maximize this function; in other words it tries to maximize the output of the discriminator for its negative instances. Furthermore, it uses a discriminator loss $\mathbf{D}(\mathbf{x}') - \mathbf{D}(\mathbf{x})$ where D tries to maximize WD defined as the difference between $D(x)$ and $D(x')$; in other words it tries to maximize the difference between its output on positive instances and its output on negative instances [4].

The training data of D belongs to two groups, the original data being denoted by x and the synthetic data by x' which is generated by G. During the training of D, x and x' are used as positive and negative instances respectively. It is important to note that during the training of D, G is semi-suspended, i.e. its weights are kept constant, but at the same time G remains to generate new in-

[4] x denotes original (positive) instances, x' denotes synthetic (negative) instances, z is a random Gaussian noise. $D(x)$ and $D(x')$ are discriminator's outputs for original and synthetic instances respectively, and $G(z)$ is the generator's output

stances to feed D with more training data. The stopping criterion happens when the critic cannot distinguish between original and synthetic samples; the output of the critic for negative samples is as high as for positive samples, making WD very close to 0. When GEG training is complete, an unlabeled synthetic dataset will be available. For all data belonging to this set, a class label is automatically added due to the way GEG is trained (class-based data restriction). This greatly facilitates the process of identifying class label for the synthetic data by avoiding the additional steps of classifying the unlabeled instances with a semi-supervised or supervised approach.

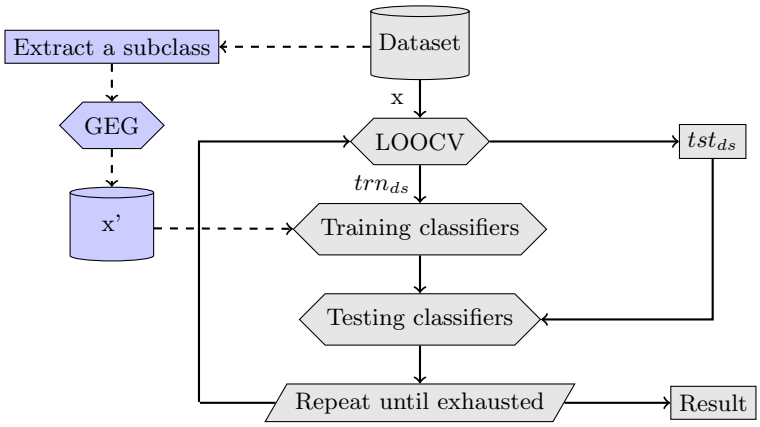

Fig. 3: Proposed LOOCV training diagram

3.2 Proposed LOOCV training diagram

Before starting the classification process, it is essential to pre-process the data. Since GED features (genes) have different ranges, this can influence the classification accuracy, resulting in misclassification of the data. The normalization step is integrated to avoid such a situation. Normalization is the process of scaling the values of numerical columns into a single range such as [0, 1] that will ensure that balanced weights are allocated to each feature. Therefore, normalization minimizes the training error, thus demonstrating the classification accuracy. The gene-level of each feature is normalized using Eq. 1:

$$new_{value} = \frac{actual_{value} - val_{min}}{val_{max} - val_{min}} \tag{1}$$

Where, val_{max} and val_{min} are the maximal and minimal original value of a given gene respectively. new_{value} is the normalized expression level. After normalization, all genes will be included between [0, 1]. As this is very small and sensitive data, the leave one out cross-validation (LOOCV) is suitable for validating model performance. LOOCV is a cross-validation technique where N data

samples are sliced into two sets: a training set trn_{ds} containing $N-1$ samples and a test set tst_{ds} containing a single sample. A classifier was trained with trn_{ds}, and the constructed model was tested with tst_{ds} by adopting any performance measurement (such as accuracy). The procedure was repeated N times, so that all instances are used once in trn_{ds} and once in tst_{ds}, but never again in both sets simultaneously. Fig. 3 shows the traditional LOOCV training process (with the gray color). Additionally, it is suggested to improve the training of classifiers by adding more instances to trn_{ds} (purple color in Fig. 3). These instances are generated using GEG to deepen the learning methods and improve the classification performance only on the original data. This means that the proposed models are only tested with the original GED.

4 Experiments, results and discussion

4.1 Datasets description

Publicly available GED [5] for colon cancer tissues and breast cancer tissues [4] was used during experimentation. These tissues are either healthy (normal histological structure tissues) or belong to one of the subtypes of cell dysplasia (cancerous tissues), for a better comprehension of gene datasets [14]. As described in Table 1, the datasets had expression levels of 6500 and 24,481 genes measured from 62 and 99 patients respectively. Both datasets are binary classification datasets. Refined (filtered) datasets were adopted instead of original datasets as an alternative to feature selection (the refined datasets contain only important genes that have a high impact factor for GED classification).

Table 1: Description of datasets used for the experiments

Dataset	Nbr of original genes	Nbr of used genes	Classes distribution
Colon cancer	6500	2000	40/22
Breast cancer	24481	4997	44/34

- **Colon cancer dataset** is composed of 40 different types of dysplasia colon tumors and 22 normal cell and histological structure of tissue samples from an Affymetrix oligonucleotide array of 6500 genes. A clustering algorithm revealed wide consistent patterns that suggest a high degree of organization underlying gene expression in these tissues [1]. The filtered dataset has 2000 gene expressions (instead of the original 6500 genes).
- **Breast cancer dataset** consists of 99 tumor samples from breast cancer patients with an initial gene count of 24,481 genes which decreased to 7,650

[5] DNA microarray data: https://homes.di.unimi.it/~valentini/DATA/MICROARRAY-DATA/

genes in [24]. Based on the estrogen receptor (ER), cancer tumors can be divided into two subgroups. Subjects with $ER+$ tumors have a better survival rate than those with $ER-$ because patients with $ER+$ tumors can benefit from anti-estrogens, such as tamoxifen. Since breast cancer is highly correlated with ER, only those genes that were linked to ER are considered, so about 5,000 genes were retained. Of the 99 patients, only 78 were considered. Of these 78 patients, 34 had a poor prognosis and were therefore labeled as $ER-$ patients, while 44 had a good prognosis, making them $ER+$ patients.

4.2 Used methods

The baseline results (simple classifiers without data augmentation) were compared to classifiers using GEG as a data augmentation technique to study the effect of GEG on classification accuracy. The classifiers used are Support Vector Machine (SVM), K-nearest Neighbors (KNN) and Decision Trees (DT) [25]. Supervised learning methods were applied as cancer datasets are labeled. The Sklearn library [19] was used for the implementation of the algorithms, with default hyper-parameters; where, SVM with a linear kernel and $C = 1$, KNN with $K = 5$, and DT with $max\_depth = 2$. It's important to note that for all experiments performed with GEG, the synthetic samples were only used during the training stage as additional training data and not as test data. Table 2 provides details of the samples generated by class.

Table 2: Number of samples generated per class for each dataset using GEG

Dataset	**Original C1/C2**	**Generated C1/C2**	**Total samples**
Colon cancer	40/22	10/28	100
Breast cancer	44/34	6/16	100

4.3 Evaluation metric

The classification accuracy was adopted as an evaluation metric to assess the performance of GEG. Accuracy is a good metric in this case because for each dataset, every classifier is trained with a balanced dataset (original data + generated data by a data augmentation technique). Accuracy is defined as the ratio between the total number of correctly classified instances and the total number of instances. Formally, accuracy can be calculated using Eq. 2:

$$Accuracy = \frac{TP + TN}{Total} \tag{2}$$

Where, True Positive (TP) (resp. True Negative (TN)) are correctly classified positive (resp. negative) samples by the classifier; False Positive (FP)(resp. False

Negative (FN)) are incorrectly classified negative samples as positive (resp. positive samples as negative); and Total ($Total = TP + FP + FN + TN$) is the total number of samples.

In ML, the dataset is usually divided into a training set and a test set. The cross-validation was used to give a more accurate estimate of a model's performance. Fig. 3 shows the LOOCV training used for the experiments, where the result of each iteration is the accuracy. The comparative study between the proposed models and the baseline was carried out using the following formula:

$$Model\,accuracy\ \% = 100 \times \frac{1}{N} \times \sum_{i=1}^{N} A_i \tag{3}$$

Where, N is the number of samples and A_i is the calculated accuracy of iteration i, multiplied by 100 to reflect model accuracy in %.

4.4 Classification results

Several classification tests were conducted using different classifiers with different training approaches. In each case, only original data (OD) was used without data augmentation and synthetic data generation using the proposed GEG (proposed approach).

Table 3: Summary of experimental results based on classification accuracy

Dataset	Method used	SVM	KNN	DT
Colon cancer	OD	87.10	82.26	75.81
	GEG	**88.71**	**83.87**	**83.87**
Breast cancer	OD	60.26	51.28	57.69
	GEG	**75.64**	**57.69**	**73.08**

The results in bold in Table 3 show the used data augmentation, i.e. GEG improved the classification accuracy compared to the baseline (original data only). Based on the good results obtained, GEG can be considered as a promising data augmentation technique to improve the classification accuracy. The results for each dataset can be summarized as follows:

- **Colon cancer dataset:** despite using three classifiers, GEG produced the best results in terms of classification accuracy. The highest result was achieved using GEG as a data augmentation technique and SVM as a classifier. GEG was able to achieve 88.71% accuracy and 55 out 62 as the number of correctly classified instances. The SVM results are due to the linearity of the data, which means that the data classes are linearly separable, justifyng the good initial results obtained without the usage of data augmentation. Therefore, the use of GEG gave a slightly higher result, meaning that the support vectors used by the SVM were adjusted (due to the use of synthetic GEG data).

– **Breast cancer dataset:** for this dataset, the combination of GEG and SVM was again a success. It improved the classification accuracy by more than 15% to reach the value of 75.64% and 59 out of 78 as a number of instances correctly classified compared to the baseline using only SVM. SVM correctly classified only 47 out of 78 instances. Similar to the colon cancer dataset, the improvement in the results was due to the adjustment of the support vectors, and thus the optimal hyper-plane when synthetic GEG-generated data was used during the training process.

5 Conclusion

The collection of medical data for cancer detection is costly and difficult to obtain due to privacy constraints. Since the available data show a disproportionate ratio between the number of available instances and the number of features, and since GED analysis uses only a small number of available samples, this could lead to inappropriate classification results. To this end, the use of sophisticated data augmentation approaches such as GANs with appropriate hyper-parameters could be very beneficial in this application area. As an alternative to data collection, it is suggested to generate synthetic samples and increase the amount of training data.

However, these generated instances must be very consistent with the original instances to obtain good results. The results of the classification accuracy of the duality between GEG and simple supervised learning methods are promising. The application of GAN to other datasets and its comparison with other data augmentation techniques remains to be done. Nevertheless, as a first step, it can be noted that GAN can be successfully used to produce synthetic samples that are in harmony with real samples, and not only for images, videos and text, but also for gene expression data.

Acknowledgement

This work was supported by the Telekom Innovation Laboratories (T-Labs) and the Research and Development unit of Deutsche Telekom.

The authors would like to express their deepest gratitude towards **Tsegaye Misikir Tashu** for his advice, valuable feedback, proofreading and assistance in overcoming technical problems.

Project no. ED_18-1-2019-0030 (Application domain specific highly reliable IT solutions subprogramme) has been implemented with the support provided from the National Research, Development and Innovation Fund of Hungary, financed under the Thematic Excellence Programme funding scheme.

References

1. Alon, U., Barkai, N., Notterman, D.A., Gish, K., Ybarra, S., Mack, D., Levine, A.J.: Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. Proceedings of the National Academy of Sciences **96**(12), 6745–6750 (1999)
2. Antipov, G., Baccouche, M., Dugelay, J.L.: Face aging with conditional generative adversarial networks. In: 2017 IEEE international conference on image processing (ICIP). pp. 2089–2093. IEEE (2017)
3. Berthelot, D., Milanfar, P., Goodfellow, I.: Creating high resolution images with a latent adversarial generator. arXiv preprint arXiv:2003.02365 (2020)
4. Buza, K.: Classification of gene expression data: a hubness-aware semi-supervised approach. Computer methods and programs in biomedicine **127**, 105–113 (2016)
5. Damian, A., Piciu, L., Turlea, S., Tapus, N.: Advanced customer activity prediction based on deep hierarchic encoder-decoders. In: 2019 22nd International Conference on Control Systems and Computer Science (CSCS). pp. 403–409. IEEE (2019)
6. 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)
7. Im, D.J., Kim, C.D., Jiang, H., Memisevic, R.: Generating images with recurrent adversarial networks. arXiv preprint arXiv:1602.05110 (2016)
8. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1125–1134 (2017)
9. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. pp. 1097–1105 (2012)
10. Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al.: Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4681–4690 (2017)
11. Li, C., Alvarez-Melis, D., Xu, K., Jegelka, S., Sra, S.: Distributional adversarial networks. arXiv preprint arXiv:1706.09549 (2017)
12. Li, C., Wand, M.: Precomputed real-time texture synthesis with markovian generative adversarial networks. In: European conference on computer vision. pp. 702–716. Springer (2016)
13. Li, J., Monroe, W., Shi, T., Jean, S., Ritter, A., Jurafsky, D.: Adversarial learning for neural dialogue generation. In: EMNLP (2017)
14. Lin, W.J., Chen, J.J.: Class-imbalanced classifiers for high-dimensional data. Briefings in bioinformatics **14**(1), 13–26 (2013)
15. Lu, Y., Kakillioglu, B., Velipasalar, S.: Autonomously and simultaneously refining deep neural network parameters by a bi-generative adversarial network aided genetic algorithm. arXiv preprint arXiv:1809.10244 (2018)
16. Marchesi, M.: Megapixel size image creation using generative adversarial networks. arXiv preprint arXiv:1706.00082 (2017)
17. 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 neural networks. bioRxiv p. 390153 (2018)
18. Mirza, M., Osindero, S.: Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784 (2014)

19. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research **12**, 2825–2830 (2011)
20. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434 (2015)
21. Reed, S., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., Lee, H.: Generative adversarial text to image synthesis. arXiv preprint arXiv:1605.05396 (2016)
22. Shang, C., Palmer, A., Sun, J., Chen, K.S., Lu, J., Bi, J.: Vigan: Missing view imputation with generative adversarial networks. In: 2017 IEEE International Conference on Big Data (Big Data). pp. 766–775. IEEE (2017)
23. Smith, E.J., Meger, D.: Improved adversarial systems for 3d object generation and reconstruction. In: Conference on Robot Learning. pp. 87–96 (2017)
24. Sotiriou, C., Neo, S.Y., McShane, L.M., Korn, E.L., Long, P.M., Jazaeri, A., Martiat, P., Fox, S.B., Harris, A.L., Liu, E.T.: Breast cancer classification and prognosis based on gene expression profiles from a population-based study. Proceedings of the National Academy of Sciences **100**(18), 10393–10398 (2003)
25. Taan, A., Farou, Z.: Supervised learning methods for skin segmentation classification (2020). https://doi.org/10.13140/RG.2.2.12444.51843/2
26. Vondrick, C., Pirsiavash, H., Torralba, A.: Generating videos with scene dynamics. In: Advances in neural information processing systems. pp. 613–621 (2016)
27. Wang, H., Qin, Z., Wan, T.: Text generation based on generative adversarial nets with latent variables. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining. pp. 92–103. Springer (2018)
28. Wang, Z., She, Q., Ward, T.E.: Generative adversarial networks: A survey and taxonomy. arXiv preprint arXiv:1906.01529 (2019)
29. Zhang, H.: Generative Adversarial Networks for Image Synthesis. Ph.D. thesis, Rutgers The State University of New Jersey-New Brunswick and University of Medicine and Dentistry of New Jersey (2019)
30. Zhang, Y., Bai, Y., Ding, M., Ghanem, B.: Multi-task generative adversarial network for detecting small objects in the wild. International Journal of Computer Vision pp. 1–19 (2020)
31. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE international conference on computer vision. pp. 2223–2232 (2017)