A Probabilistic Interpretation of Data Augmentations

JONATHAN COLAÇO CARR
McGill University
December 20, 2021.

In this paper we propose probabilistic framwork to categorize data augmentation strategies based on their knowledge of the machine learning environment. We provide examples of how to apply this framework to several image augmentations, and show how it explains the efficacy salience-based augmentations as well as adversarial training strategies. While we focus on examples from computer vision, our methodology applies to any supervised learning environment.

1. Introduction

Data augmentation refers to a collection of strategies to generate more samples from a dataset. As an effective way to prevent overfitting, this topic has been explored in a myriad of different contexts, reviewed in [2, 4, 16]. Most review papers about data augmentations provide taxonomies that are based on the mechanical procedure of the augmentations (such as geometric transformations vs. deep learning approaches). However, in these frameworks it remains unclear why one augmentation technique would work over another for a given problem [16].

Without consistent baselines, new augmentations strategies are often compared with arbitrary competitors, making it challenging to assess their fundamental impact. In this paper, we present a framework which classifies augmentations based on their access to information about the learning environment. This naturally leads to a fair comparison of data augmentation strategies.

1.1. *Definitions*

In order to define data augmentations, we first define the supervised machine learning problem, following [12]. We observe samples $x \in \mathcal{X}$, where $\mathcal{X}$ is the *domain*, under some distribution $\mathcal{D}$. For image classification, $\mathcal{X} \subset \mathbb{R}^{W\times H}$ or $\mathbb{R}^{W\times H\times D}$ for grayscale images. For colored images, $\mathcal{X} \subset \mathbb{R}^{W\times H\times 3}$ or $\mathbb{R}^{W\times H\times D\times 3}$. To each observation x, we associate a label $y \in \mathcal{Y} = \{1,\dots,K\}$. The *dataset*

$$S_m = \{(x_1,y_1),\dots,(x_m,y_m)\},$$

consists of m observations $(x_i,y_i) \in \mathcal{X}\times\mathcal{Y}$. Given a loss function $\ell : \mathcal{Y}\times\mathcal{Y} \to \mathbb{R}$, Our objective in training a model is to minimize the *empirical loss*,

$$L_{S_m}(f) = \frac{1}{m}\sum_{i=1}^{m} \ell(f(x_i,w),y_i).$$

Typically, the functions considered are restricted to a hypothesis class

$$\mathcal{H} = \{f(x,w) :\ w \in \mathbb{R}^D\}.$$

In deep learning models, $D \gg m$. Thus, in minimizing $L_{S_m}[f]$, the model f may fall victim to *overfitting*, i.e. memorizing irrelevant features in the training set. This effect can be quantified via generalization error.

Definition 1.1. Given a model $f \in \mathcal{H}$, a training set $\mathcal{S}_m = \{(x_1, y_2), \ldots, (x_m, y_m)\}$ and a test set $\mathcal{T} = \{(x'_1, y'_1), \ldots, (x'_l, y'_l)\}$, we define the *empirical generalization error of f* as

$$GE(f) = L_{\mathcal{T}}[f] - L_{\mathcal{S}_m}[f]. \tag{GE}$$

The strongest assumption we can make is that $\mathcal{S}_m$ and $\mathcal{T}$ are sampled i.i.d. from the data distribution $\mathcal{D}$. While this is a standard assumption in statistical learning theory, it is often not the case in practice. In fact, many data augmentation techniques try to account for a discrepency between the distributions of $\mathcal{S}_m$ and $\mathcal{T}$.

Importantly, generalization error is defined with respect to a model f. Once a sufficient hypothesis class $\mathcal{H}$ is chosen, a reasonable goal would be to reduce the *generalization error* of our model. With no prior knowledge of $\mathcal{T}$, the generalization error is a good indication of how well a model perform on examples it has not seen before.

2. Augmentations

Over decades of research, data augmentations have evolved from simple geometric transformations of existing training samples [3, 9], to complex functions of the hypothesis space, loss function, or even other deep learning models[8]. The tremendous scope of this term motivates the following abstract definition.

Definition 2.1. An *augmentation A* is a random variable that takes values in $\mathcal{X} \times \mathcal{Y}$.

It is not clear *a priori* which augmentations, if any, will aid in minimizing (GE). Indeed, the challenge is to choose a family of augmentations which has this effect. From the definition of (GE), it is clear that our augmentations should be based on what may be assumed about the learning environment. This information can be encoded in the domain space of the random variable A.

2.1. *Model-Agnostic Augmentations*

The first type of information that might exist in the domain space of A is hypothesis class $\mathcal{H}$.

Definition 2.2. An augmentation A is *model-agnostic* if the behaviour of A does not depend on the hypothesis class $\mathcal{H}$. I.e. for any two hypothesis classes $\mathcal{H}_1, \mathcal{H}_2$,

$$\mathbb{E}\{A\} = \mathbb{E}\{A\mathbf{1}_{\mathcal{H}=\mathcal{H}_1}\} = \mathbb{E}\{A\mathbf{1}_{\mathcal{H}=\mathcal{H}_2}\}.$$

One obvious advantage of model-agnostic augmentations is that they can be employed before a hypothesis class $\mathcal{H}$ is established. We illustrate our definition of model-agnostic augmentations below. Each example lead to significant performance increases on the state-of-the-art models in its era on popular datasets (e.g. ImageNet, CIFAR-10, etc.).

Example 2.1. *Noise Injections* [11] consist of injecting an image with random values. Given $x \in \mathcal{X}$ some distribution $\mathcal{D}$ (usually Gaussian or uniform), we define a noise injection as

$$A : (\mathcal{X} \times \mathcal{Y}, \mathcal{D}) \longrightarrow \mathcal{X} \times \mathcal{Y}$$
$$A((x,y),d) = (x \oplus d, y)$$

where $\oplus$ is the pointwise addition to each pixel of x.

Example 2.2. *The Cutmix algorithm* [19]. In this strategy, patches of examples are cut and pasted among training images where the ground truth labels are also mixed proportionally to the area of the patches. In detail, given two points (x_A, y_A) and $(x_B, y_B) \in \mathbb{R}^{W \times H}$,

$$A : (\mathcal{X} \times \mathcal{Y})^2 \longrightarrow \mathcal{X} \times \mathcal{Y}$$
$$A((x_1,y_1),(x_2,y_2)) = (M \odot x_A + (1-M) \odot x_B, \lambda y_A + (1-\lambda) y_B),$$

where $\lambda \sim \text{Unif}(0,1)$ $M \in \{0,1\}^{W \times H}$ is a binary image with the proportion of 1s equal to λ.

While model-agnostic augmentations assume less information about the learning environment, they are not necessarily less complex.

Example 2.3. *Style Randomization* [8] randomizes texture, contrast, and color of training samples while preserving its semantic shape and content. The explicit details of the style transfer algorithm are beyond the scope of this paper, but a full treatment can be found in [5]. The augmentation A is a highly non-linear function of the input image, which is conditioned to a neural style algorithm. However, it is important to note that A acts independently of the hypothesis class $\mathcal{H}$. A sample of the style randomization is shown in Figure 1.

Figure 1. Style augmentation applied to an image x, adapted from [8].

2.2. *Model-Dependent Augmentations*

For machine learning tasks with a well established hypothesis class, model-dependent augmentations have been shown to drastically increase the performance of model-dependent augmentations [1, 7, 10, 16].

Definition 2.3. An augmentation A is *model-dependent* if it is not model agnostic.

Model-dependent augmentations use the hypothesis class to find an initial well-trained function. Assuming $\mathcal{S}_m$ is a faithful representation of the test set $\mathcal{T}$, one can find a near optimal function f by minimizing

the empirical loss on the original dataset $\mathcal{S}_m$, before augmenting it with samples. The well-trained model is made precise by the following.

Definition 2.4. A model is ε-*well trained* if $GE(f) < \varepsilon$.

Model-dependent augmentations exploit the existence of a well-trained model to guide the choice of augmentations $\mathcal{A}$. We provide two examples of model-dependent augmentations below which, in both cases, enhance model-agnostic augmentations using a well-trained model f.

2.2.1. Salience-Based Augmentations
One of the ways in which augmentations make use of well-trained functions is through salience maps [17].

Definition 2.5. Given an input image $x_0 \in \mathcal{X}$, a class $y_0 \in \mathcal{Y}$, and a scoring function $s : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$, we define the *salience map of* x_0 *with respect to class* y_0 as

$$g(x_0, y_0) = |\nabla_x s(f(x_0), y_0)|.$$

Intuitively, these maps indicate which pixels of x_0 need to change the least in order for the class score y_0 to change the most. Salience can be used to guide 'random swapping' procedures (such as the Cutmix augmentation), which may inadvertently swap important features onto images with different labels.

Example 2.4. The *SalienceMix* [15] augmentation avoids information loss by predefining the importance of a given region using the salience map of a well-trained model. In particular for a region $\chi \subset x_0$, an *importance score* is defined as

$$I(\chi, x, y) = \sum_{(i,j)\in\chi} g_{ij}(x, y).$$

For a fixed a threshold $\tau > 0$, augmentations on the sample x are only performed on regions χ for which $I(\chi, x, y) < \tau$. Using this importance score as a condition for various random swapping/erasing techniques (such as the *Cutmix* algorithm defined above) was shown to increase the efficacy of these augmentations.

Similar work [6, 14] shows that access to a salience maps avoids the issue of inadvertently creating images which are mislabelled.

2.2.2. Adversarial Examples
Adversarial examples augmented samples that have been carefully perturbed in order to incur a misclassification by the model (see Figure 2). The expected behaviour of adding adversarial examples - to make the model more robust to small perturbations - is akin to the *Noise Injection* method describe above. These augmentations can be used to delay overfitting memorization much better than explicit regularization techniques (such as dropout) or Gaussian noise injections [1, 18].

Remark 2.1. Adversarial training is a highly active area of research [2, 7, 10, 13] which extends beyond our definition of augmentations. Some techniques may take other factors of the learning environment into account such as the loss function and optimization strategy.

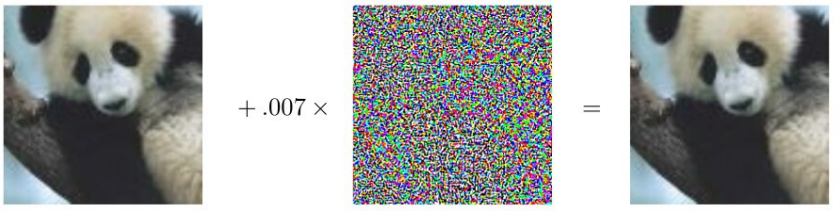

Figure 2. Adversarial example, adapted from [7], shows the *fast gradient sign method* for finding adversarial examples. In this case, a noise vector is added whose components are equal to the sign of the components of the gradient of the loss function J. This disturbance, imperceptible to the human eye, causes GoogLeNet [18] to misclassify examples from ImageNet.

3. Discussion

The distinction between model-dependent and model-agnostic augmentations allows for a more reasonable comparison between data augmentation strategies. While new algorithms such as *SalienceMix* are effective in reducing generalization error, it should not come as a surprise that it outperforms *Cutmix* augmentation in this regard, since it has access to the hypothesis class in question. While the use of adversarial examples is significantly more effective than random noise injections in preventing data memorization, adversaries are predicated on the existence of a well-trained function, which may not always be assumed.

Model-agnostic and model-dependent augmentations can be further subdivided based on other assumptions and information about the learning environment, such as the distribution of the training set $\mathcal{S}_m$ with respect to the ambient distribution $\mathcal{X}$, or even knowledge of the optimization strategy, as is the case with some adversarial training techniques. In future work we hope to expand upon this classification system as well as provide rigorous bounds for increasing generalization performance. We expect that we can guarantee the superiority of model-dependent augmentations over model-agnostic ones in reducing (GE), although this has not been proven yet.

4. Conclusion

While the space of all augmentations is vast and expanding rapidly, our classification of augmentations, based on their assumptions about the machine learning problem, allows one to compare augmentation strategies in a more equitable way. Hopefully a better-regulated arena of data augmentation strategies will lead to a systematic way of reducing generalization error.

References

1. D. Arpit, S. Jastrzębski, N. Ballas, D. Krueger, E. Bengio, M. S. Kanwal, T. Maharaj, A. Fischer, A. Courville, Y. Bengio, and S. Lacoste-Julien. A Closer Look at Memorization in Deep Networks. *arXiv e-prints*, page arXiv:1706.05394, June 2017.
2. T. Bai, J. Luo, J. Zhao, B. Wen, and Q. Wang. Recent Advances in Adversarial Training for Adversarial Robustness. *arXiv e-prints*, page arXiv:2102.01356, Feb. 2021.

3. Y. Bengio, F. Bastien, A. Bergeron, N. Boulanger–Lewandowski, T. Breuel, Y. Chherawala, M. Cisse, M. Côté, D. Erhan, J. Eustache, X. Glorot, X. Muller, S. Pannetier Lebeuf, R. Pascanu, S. Rifai, F. Savard, and G. Sicard. Deep learners benefit more from out-of-distribution examples. In G. Gordon, D. Dunson, and M. Dudík, editors, *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pages 164–172, Fort Lauderdale, FL, USA, 11–13 Apr 2011. PMLR.
4. S. Chen, E. Dobriban, and J. H. Lee. A Group-Theoretic Framework for Data Augmentation. *arXiv e-prints*, page arXiv:1907.10905, July 2019.
5. L. A. Gatys, A. S. Ecker, and M. Bethge. A Neural Algorithm of Artistic Style. *arXiv e-prints*, page arXiv:1508.06576, Aug. 2015.
6. C. Gong, D. Wang, M. Li, V. Chandra, and Q. Liu. KeepAugment: A Simple Information-Preserving Data Augmentation Approach. *arXiv e-prints*, page arXiv:2011.11778, Nov. 2020.
7. I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and Harnessing Adversarial Examples. *arXiv e-prints*, page arXiv:1412.6572, Dec. 2014.
8. P. T. Jackson, A. Atapour-Abarghouei, S. Bonner, T. Breckon, and B. Obara. Style Augmentation: Data Augmentation via Style Randomization. *arXiv e-prints*, page arXiv:1809.05375, Sept. 2018.
9. Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
10. B. Li, S. Wang, S. Jana, and L. Carin. Towards Understanding Fast Adversarial Training. *arXiv e-prints*, page arXiv:2006.03089, June 2020.
11. F. J. Moreno-Barea, F. Strazzera, J. M. Jerez, D. Urda, and L. Franco. Forward noise adjustment scheme for data augmentation. In *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 728–734, 2018.
12. A. M. Oberman. Partial differential equation regularization for supervised machine learning. *arXiv e-prints*, page arXiv:1910.01612, Oct. 2019.
13. N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. Berkay Celik, and A. Swami. Practical Black-Box Attacks against Machine Learning. *arXiv e-prints*, page arXiv:1602.02697, Feb. 2016.
14. D. V. Ruiz, B. A. Krinski, and E. Todt. Anda: A novel data augmentation technique applied to salient object detection. In *2019 19th International Conference on Advanced Robotics (ICAR)*, pages 487–492, 2019.
15. A. F. M. Shahab Uddin, M. Sirazam Monira, W. Shin, T. Chung, and S.-H. Bae. SaliencyMix: A Saliency Guided Data Augmentation Strategy for Better Regularization. *arXiv e-prints*, page arXiv:2006.01791, June 2020.
16. C. Shorten and T. M. Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1):60, Jul 2019.
17. K. Simonyan, A. Vedaldi, and A. Zisserman. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. *arXiv e-prints*, page arXiv:1312.6034, Dec. 2013.
18. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going Deeper with Convolutions. *arXiv e-prints*, page arXiv:1409.4842, Sept. 2014.
19. S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo. CutMix: Regularization Strategy to Train Strong Classifiers with Localizable Features. *arXiv e-prints*, page arXiv:1905.04899, May 2019.