Climate Adaptation: Reliably Predicting from Imbalanced Satellite Data

Ruchit Rawal
Netaji Subhas University of Technology,
New Delhi, India.
ruchitr.ec.17@nsit.net.in

Prabhu Pradhan
Max Planck Institute for Intelligent Systems,
Tübingen, Germany.
prabhu.pradhan@tuebingen.mpg.de

Abstract

The utility of aerial imagery (Satellite, Drones) has become an invaluable information source for cross-disciplinary applications, especially for crisis management. Most of the mapping and tracking efforts are manual which is resource-intensive and often lead to delivery delays. Deep Learning methods have boosted the capacity of relief efforts via recognition, detection, and are now being used for non-trivial applications. However the data commonly available is highly imbalanced (similar to other real-life applications) which severely hampers the neural network’s capabilities, this reduces robustness and trust. We give an overview on different kinds of techniques being used for handling such extreme settings and present solutions aimed at maximizing performance on minority classes using a diverse set of methods (ranging from architectural tuning to augmentation) which as a combination generalizes for all minority classes. We hope to amplify cross-disciplinary efforts by enhancing model reliability.

1. Introduction

The last decade has witnessed tremendous growth both in computational power and scientific methods for pattern recognition and data science. Machine Learning is a tool driving many technologies across diverse sectors. However the fuel that drives this growth is data, and as is with every fuel it’s not directly usable. A critical problem is class imbalance, both in supervised and unsupervised form of learning algorithms. A dataset can be treated as imbalanced if there is a noticeable mismatch between the target variable and other values. For example, medical-diagnostics data is conventionally biased towards the negative class. Other examples are in fraud detection, NLP classification, visual recognition, etc. Experimentally, high instability in performance metrics has been observed in vanilla models tested on imbalanced datasets [3].

Commonly, deep-net models are built to maximize predictive accuracy (ex. classification) but this metric is uneventful for the cases with limited labels, extreme classification etc. [26]. This happens because the trained classifier focuses only on the most-numerous class (since it has a higher proportion) while remaining below-par on minority classes. This may prove catastrophic in critical use cases like medical diagnostics and self-driving cars where the rare instances are of utmost importance.

Our use case consists of satellite imagery of African region which is labelled to help automate the process of predicting drought, cattle sustenance etc. via estimating the quality of forage [15]. Usually, non-profit organizations cannot employ a dedicated team of ML engineers/researchers or clusters of GPUs [7], models that can perform robustly and reliably at low requirements can be pragmatically utilized by domain-experts and local administration [29].

Presently, researchers tend to tackle the imbalance issues (either at input or intermediate pipeline) in its narrow context with domain-specific solutions. We present drawn-out insights on several techniques to mitigate data-imbalance problems. The contributions of this paper are:

- We use a deep generative model for synthetic data augmentation of multi-spectral images. To the best of our knowledge, this specific area is still unexplored.
- We show that certain spectral bands are better for particular tasks (here, vegetation area analysis).
- We show that a combination of Cyclic Learning Rate (CLR) [38] + Stochastic Weight Averaging (SWA) [20] is suitable for extreme imbalance scenarios.
- We further cement the compatibility of LDAM: Label-distribution aware loss-function [5], which works better than crude re-sampling and can be further improved by using class-balanced loss [6].

The rest of this paper is as follows: Section 2 introduces the dataset. Section 3 provides details on our modifications.
to the base neural-net model. It also includes subsections on Loss Function (3.1) which gives an overview of sampling functions, Cyclic Learning Rate (3.2) which is a popular training routine, and Stochastic Weight Averaging (3.3) as a powerful regularizer for handling data-imbalance issues. Section 4 provides details on multi-spectral imagery from the lens of machine learning. Section 5 presents our experiments with synthetic data augmentation followed by our overall results. Finally we conclude the paper in Section 7 with a short discussion on performance metrics.

We use intra-class variance (ICV), Balanced accuracy [2] and Recall as performance metrics (definitions in Sec. 1.1).

We present all our observations in two graph plots - Figure 2 (ValAcc vs ICV) and Figure 3 (BalAcc vs ICV).

Our codebase will be made publicly available.

1.1. Performance Metrics

As we see in [26, 39, 32, 1], accuracy is not the best metric to evaluate imbalanced datasets, as it can be very misleading. Metrics that provide better insights [34] include:

- **Recall**: Recall portrays the fraction of true positives could be detected correctly. It is defined as \( \frac{\text{TruePositive}}{\text{TruePositive} + \text{FalseNegative}} \). Thus a low recall signifies a high number of false negatives which is undesirable in a real-world setting.

- **Balanced Accuracy (BalAcc)**: The arithmetic mean of the TPR (True Positive Rate) and TNR (True Negative Rate). Thus if the model is exploiting class-imbalanced problem to increase the vanilla accuracy, the balanced accuracy will drop significantly and reflect the poor performance.

- **Intra-Class Variance (ICV)**: \( \sum_{i=0}^{4} (acc - class_i)^2 \) (where \( acc \) denotes the validation accuracy and \( class_i \) denotes the accuracy of the \( i^{th} \) class for a given experiment) the main rationale is expose the models which have high variance amongst the per class-accuracies owing to overfitting on the frequent class in comparison to robust models i.e. low variance

Table 8 presents a comparative list of our final results as per the aforementioned benchmarks.

2. Data-set

The expert-labelled, multi-spectral satellite (LANDSAT) data [15] was released as a bid to enhance drought detection pipelines. It essentially consists of 100,000 images split into 86,317 training and 10,778 validation images, having a spatial resolution of 65x65 pixels over 10 spectrum bands. Each image is labeled by a human expert as- the number of cows the geographical location at the center of the image can support, serving as a measure of forage quality of the location and further as an indicator of whether the location is arid (drought-hit).

The dataset is highly imbalanced (roughly 60% of the data gathered is of class 0, classes 1 and 2 have 15% each, and the remaining 10% is class 3). The model can erroneously achieve 60% accuracy just by predicting 0 every time. However, such high mis-classification is very problematic since these algorithms will be deployed in high-stake real-world settings. We would like to make dense predictions no matter the location of the pixel, since there is high amount of sparsity in the labels. Hence, we need to train a model that is satisfactorily robust to out-of-distribution (o.o.d) samples and generalizes well on all the inherent classes i.e. independent-&-identically-distributed (i.i.d) samples. We focus on striking a pragmatic balance.

3. Network Architectures

Ever since winning the 2015 ILSVRC [33] challenge ResNet [14] has inspired a family of deep convolutional neural networks. The skip connections in ResNet allow one to build deep networks (up to 1000 layers) while still keeping them optimized. He et. al [14] demonstrated that even for fixed baseline architecture increase in depth almost always leads to increased accuracy.

While Scaling in depth [14] is the go-to method to boost a network’s accuracy, other less popular scaling methods include scaling by width [44] and resolution [19]. Tan et. al [41] in their work showed that while scaling (in width, depth, resolution) improves model accuracy, the accuracy saturates after a certain level. They argued that different scaling dimensions (height, width, resolution) are not independent and the key to successfully scale deep networks is in balancing scaling in different dimensions rather than scaling in one direction only. To harmonize the scaling in all dimensions they proposed a compound scaling method which utilized \( \phi \) (compound scaling coefficient) to uniformly scales the network’s depth, width and resolution.

\[
\text{Depth} : d = \alpha^\phi, \quad \text{Width} : w = \beta^\phi, \quad \text{Resolution} : r = \gamma^\phi
\]

s.t. \( \alpha + \beta^2 + \gamma^2 \approx 2 \quad \alpha, \beta, \gamma \geq 1 \)

However, scaling doesn’t change the core layer operations making it imperative to have a solid baseline network for achieving desired outcomes, Tan et al. [41] leveraged Neural Architecture Search [47] to propose a new baseline “Efficient-Net” by optimizing for both accuracy and FLOPS.

In Table 1 we give a baseline for ResNet-50 and Efficient-Net B4. We also apply standard data augmentation e.g. Random Horizontal-Flips, Random Vertical-Flips and Random Rotation after normalizing the data.
3.1. Loss Function and Sampling

Deep learning networks for all their might still fare very poorly on highly imbalanced datasets. Re-sampling and re-weighting are the most common techniques used to cope with class imbalance problem.

1. Re-sampling:

   (a) Oversampling [37, 45, 3, 4]: Augmenting the dataset with multiple copies of minority class samples, however since we inherently have low information about the minority class oversampling more often than not leads to overfitting on minority class [6].

   (b) Undersampling [13, 21, 3]: Undersampling is achieved by rejecting samples from the more-frequent classes. Since we are loosing out on purpose in order to equalize the class-count, undersampling technique aren’t possible in case of high class imbalance [6].

2. Re-weighting [17, 18]: Different set of weights ($\propto \frac{1}{n_j}$, where $n_j = \text{total samples of } j^{th} \text{ class}$) are assigned to different classes. However re-weighting techniques cause instability in network’s optimization under extreme class imbalance [6].

Both re-sampling and re-weighting conclusively aim to augment the training distribution to become much more identical to the test distribution. However, due to the aforementioned flaws performance of minority class is generally increased on the cost of the network’s ability to learn the majority class well.

Cao et al. [5] designed a label-distribution aware loss function (LDAM) that regularizes the minority class much more strongly than the majority class, motivating the network to improve generalization on the minority class without suppressing the network’s ability to learn the majority class. Strong regularisation here can be understood in terms of enforcing bigger margins for the minority class as compared to the majority class. Moreover this approach is orthogonal to re-weighting and re-sampling, ensuring flexibility depending on level of imbalance in one’s dataset.

In the same work, Cao et al. [5] proposed a “deferred re-balancing training” procedure which divides the training procedure into two stages. The first stage uses Empirical Risk Minimization with LDAM loss, learning a good initial representation. The second stage employs re-weighted LDAM loss with a smaller learning rate. The main rationale behind this is to bypass the problems caused by re-weighting in the optimization process of a Neural Network by first learning a good initial representation and then optimizing on that. We also employ a re-sampling scheme ($\propto \frac{1}{n_j}$) orthogonal to the LDAM+DRW routine. Table 2 presents our results with LDAM.

In the next subsection we give a brief on Cyclic Learning Rate (CLR) and its advantages.

### 3.2. Cyclical Learning Rates (CLR)

Learning rate is responsible for scaling the gradients at each weight update and is one of the most important hyperparameters to tune while training a deep neural network as too small a learning rate will encourage very small steps and hence the network might not converge at all, whereas too high a learning rate will propel divergent behavior. The optimal learning rate depends on the network’s loss surface and usually is not feasible to calculate.

The cyclical learning rate [38] oscillates between a range of values, going against the conventional wisdom of exponentially/step-wise decreasing the learning rate as training progresses. The advantages of doing this are -
1. Stuck on a sharp minimum [25] - Networks with flatter minima tend to be more robust than the ones with sharp minima, as flatter minima ensure that we are in optimal minima region in the test loss surface as well and hence generalize better, periodically increasing the value of learning rate will help to get out of the sharp minima more quickly.

2. Stuck on saddle points [22, 38] - When training a Deep network it is very likely that the loss surface topology contains a lot of saddle points. Thus having per periodic boost of high learning rate is very useful as it helps in traversing the saddle points more quickly (since the gradient value is already very low here).

The next section is on Stochastic Weight Averaging (SWA) that is a very promising regularization technique and we outlay the setup details and it’s benefits for our problem.

3.3. Stochastic Weight Averaging (SWA)

Another go-to methodology machine learning practitioners generally adopt while training models is ensemble learning. Ensemble learning improves predictions by combining [for example voting, averaging etc] results of various models. However when training Deep Neural Networks it is not possible to train multiple models on the dataset due to time and compute constraints.

Garipov et al. [9] in their work on Fast Geometric Ensembles showed that using cyclical learning rates with stochastic gradient descent traversed on the periphery of the optimal weights but never quite reached it’s center. They selected the network with weights on the periphery to form the ensemble. This helped in training the ensemble in time required to train one network.

Stochastic Weight Averaging [20] uses the same setup i.e. high frequency cyclical/constant learning rate with SGD to traverse around the optimal weight set, and then does averaging in the weight domain only at different snapshots of training. This allows weights to reach the much desired optimal set. The advantages of this are following:

1. Faster inference time compared to Garipov et al. [9], as we only have one model as the end result, compared to waiting for k results from k models.

2. Given that the underlying data distribution is the same, it is fair to assume that the test and train datasets will have similar loss surfaces. Thus it makes much more sense to aim for a more flatter minima while training than a sharp one [even if it leads to higher training error], as it will ensure that we are in an optimal minima region in the test loss surface as well, leading to a more robust network.

We find that using SWA in combination with Adam optimizer and the CLR setup we were able to significantly improve the low/mid class accuracy and subsequently train a more robust network Table 4.

In the next section we present our brief insights linking the remote sensing community with the machine learning engineers. The bands are a key component and must be studied in more detail for better cross-linking when being used with neural networks.

4. Training On Subset of Bands

Multi-spectral Images (MSI) are described by 3 to 10 narrow spectral bands. This high spectral information is very beneficial as by combining different spectral bands we can infer different information, leading up to terabytes of data produced per day.

Since adjacent bands in MSI are highly correlated, there is a lot of redundancy in our data. This contrary to conventional wisdom, leads to degradation of accuracy on increasing the number of bands in MS images [12], also using too many spectral bands incur high computational cost as well as more inference time.

Thus it makes sense to use only those spectral bands which motivate the network to learn better feature representations for separating specific classes. The selected band

<table>
<thead>
<tr>
<th>Model</th>
<th>Training Details</th>
<th>Validation Accuracy</th>
<th>Recall 0 1 2 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Efficient-Net B4</td>
<td>Learning Rate: CLR</td>
<td>0.7002 0.8089 0.9199 0.9407 0.9407</td>
<td></td>
</tr>
<tr>
<td>Stochastic Weight Averaging: Yes</td>
<td>Learning Rate: CLR</td>
<td>0.7022 0.7792 0.5739 0.5409 0.7436</td>
<td></td>
</tr>
<tr>
<td>Bands: All</td>
<td>Sample: ∝</td>
<td>0.7047 0.7996 0.5536 0.5475 0.6428</td>
<td></td>
</tr>
<tr>
<td>Stochastic Weight Averaging: Yes</td>
<td>Learning Rate: CLR</td>
<td>0.7022 0.7792 0.5739 0.5409 0.7436</td>
<td></td>
</tr>
<tr>
<td>Bands: 4,3,2</td>
<td>Sample: ∝</td>
<td>0.7047 0.7996 0.5536 0.5475 0.6428</td>
<td></td>
</tr>
</tbody>
</table>

Table 4: SWA Experiment

| Bands: 5,4,3 | Sample: ∝ | 0.7047 0.7996 0.5536 0.5475 0.6428 |
| Stochastic Weight Averaging: Yes | Learning Rate: CLR | 0.7022 0.7792 0.5739 0.5409 0.7436 |
| Bands: 6,5,2 | Sample: ∝ | 0.7047 0.7996 0.5536 0.5475 0.6428 |

Table 5: Using subset of bands
performance is often conditional on many aspects of the classification pipeline such as the nature of the adopted classifier and its parameter configurations [40].

A major hurdle was deciding the importance of each spectral band, since there is not a lot of literature specific to neural networks. We experimented with three different band combinations based on their characteristics. [8]

1. 4-3-2: **Natural Color**  This band combination results in the image appearing as perceived by the human eye.

2. 5-4-3: **Near Infrared Composite**  This combination contains near-infrared(5), red(4), green(3) bands. This combination is particularly useful while analyzing vegetation, crops and wetlands as it is able to capture the near-infrared light reflected by chlorophyll.

3. 6-5-2: **Agriculture**  It is a combination of SWIR-1 (6), near-infrared (5) and blue (2). The short-wave and near infrared allows this combination to be used for crop monitoring.

As observed in Table 5, the combination 6-5-2 seems to work the best for the given dataset.

We believe the original dataset is small but inherently complex due to overlap of several spectral bands and thus data augmentation is very beneficial. The next section expands on the data generation component of our project.

5. **Generating Synthetic Images**

The introduction of Generative Adversarial Networks (GANs) [11] sprung up many exciting research directions, the field has grown steadily with numerous applications in image super-resolution, in-painting, image-to-image translation, image enhancement (For example, earth observation/remote sensing [27, 42]).

<table>
<thead>
<tr>
<th>Model</th>
<th>Training Details</th>
<th>Validation Accuracy</th>
<th>Recall 0</th>
<th>Recall 1</th>
<th>Recall 2</th>
<th>Recall 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Efficient-Net B4</td>
<td>Learning Rate : 1e-5</td>
<td>Loss : Cross-Entropy</td>
<td>0.76</td>
<td>0.9199</td>
<td>0.4114</td>
<td>0.5739</td>
</tr>
<tr>
<td></td>
<td>Dataset : Original</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Efficient-Net B4</td>
<td>Learning Rate : 1e-5</td>
<td>Loss : Cross-Entropy</td>
<td>0.67</td>
<td>0.6815</td>
<td>0.6619</td>
<td>0.6258</td>
</tr>
<tr>
<td></td>
<td>Stochastic Weight Averaging : No</td>
<td>Dataset : GAN-Augmented</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Efficient-Net B4</td>
<td>Learning Rate : CLR</td>
<td>Loss : Cross-Entropy</td>
<td>0.74</td>
<td>0.8156</td>
<td>0.5941</td>
<td>0.6138</td>
</tr>
<tr>
<td></td>
<td>Stochastic Weight Averaging : Yes</td>
<td>Dataset : Original</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Efficient-Net B4</td>
<td>Learning Rate : CLR</td>
<td>Loss : Cross-Entropy</td>
<td>0.70</td>
<td>0.7459</td>
<td>0.5842</td>
<td>0.5840</td>
</tr>
<tr>
<td></td>
<td>Stochastic Weight Averaging : Yes</td>
<td>Dataset : GAN-Augmented</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 6: GAN Augmented Dataset Comparison

Table 7: Training Details for GAN

<table>
<thead>
<tr>
<th>Training Details</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resolution</td>
<td>64x64</td>
</tr>
<tr>
<td>Epochs</td>
<td>45</td>
</tr>
<tr>
<td>Learning Rate</td>
<td>2e-4</td>
</tr>
<tr>
<td>$\beta_1$</td>
<td>0.5</td>
</tr>
<tr>
<td>$\beta_2$</td>
<td>0.999</td>
</tr>
</tbody>
</table>

Standard data augmentation has been used as a go-to technique for enhancing generalizability. Generative adversarial networks offer a novel method for data augmentation [35], but have still not been adopted by either the earth observation or remote sensing community. We use DCGAN [30], which employs deep convolutional neural networks for both the Generator (G) and Discriminator (D), to generate synthetic images for the low represented classes as a form of data-augmentation to equalize the number of samples of each class. We only operate on a subset of bands (6-5-2), since it is easier to critic the visual perceptibly of images this way than all the bands combined.

The main motivation behind equalizing the number of classes was to make the network learn improved discriminatory features and hence becomes more robust.

We monitored the visual perceptibly of generated images over the training period (60 epochs) and found that the network converges at about 45 epochs, see Figure 4. The final dataset (GAN-Augmented) consisted of 120,000 images with 30,000 images from each class. The architectures used for D and G are kept same as described in [30]. Training details are shown in Table 7 and Loss plots are in Figure 1.

Table 6 demonstrates the results we obtained from the network (in combination with various training methodologies) on the GAN-Augmented dataset, a significant increase in the per-class accuracies of rare-classes was observed.

Standard data augmentation has been used as a go-to technique for enhancing generalizability. Generative adversarial networks offer a novel method for data augmentation [35], but have still not been adopted by either the earth observation or remote sensing community. We use DCGAN [30], which employs deep convolutional neural networks for both the Generator (G) and Discriminator (D), to generate synthetic images for the low represented classes as a form of data-augmentation to equalize the number of samples of each class. We only operate on a subset of bands (6-5-2), since it is easier to critic the visual perceptibly of images this way than all the bands combined.

The main motivation behind equalizing the number of classes was to make the network learn improved discriminatory features and hence becomes more robust.

We monitored the visual perceptibly of generated images over the training period (60 epochs) and found that the network converges at about 45 epochs, see Figure 4. The final dataset (GAN-Augmented) consisted of 120,000 images with 30,000 images from each class. The architectures used for D and G are kept same as described in [30]. Training details are shown in Table 7 and Loss plots are in Figure 1.

Table 6 demonstrates the results we obtained from the network (in combination with various training methodologies) on the GAN-Augmented dataset, a significant increase in the per-class accuracies of rare-classes was observed.

6. **Results**

We evaluate various techniques in a combination setting to facilitate training of robust Deep Neural Networks. We provide baseline metrics for both architectures (ResNet-
50 and Efficient-Net B4) in Table 1, and observe that the baselines fall prey to overfitting owing to high class imbalance. Table 2 advocates LDAM loss as a label-dependent regularizer which leads to a reduction in Intra-class variance (ICV) and improvements in balanced-accuracy, see Figure 3. We observe that performance of SWA+LDAM+CLR (all bands, Table 8 - 6) performs considerably better than other baselines. Table 2 advocates LDAM loss as a label-dependent regularizer which leads to a reduction in ICV, see Figure 3. We observe that performance of SWA+LDAM+CLR (all bands, Table 8 - 6) performs considerably better than other baselines.

### 6.1. Limitations

- We did not incorporate adversarial training/defense.
- We did not explore alternative generative models ex. Kernel-based GANs [28], Variational Autoencoders family (VQ-VAEs [43, 31], hybrid VAE-GAN [24]).
- No class-activation mapping for model explanation or other interpretability mechanism [46, 36, 16].

### 7. Conclusion

There is a lot of focus on handling or curbing the adverse effects of imbalanced data. Mitigating class imbalance is an important research area, as it will allow trust-worthy solutions in the form of deep neural networks in many eclectic fields. As per trend, deep learning networks are tuned to maximize the total accuracy over the entire dataset, thus focusing on the majority-class samples. As a result, the models under-perform on minority class(es) samples leading to bad intra-class generalization and low robustness.

We provide a comparative overview of diverse yet latest methodologies for operating on skewed datasets that is suffering from class-imbalance problems. This diverse set of techniques ranges from discussions on state-of-the-art convolutional neural network architectures, label-dependent loss functions, learning-rate routines, generating Deep Neural Network ensembles and finally generating data samples using DC-GAN.

We conclusively aspire to serve as a toolkit for practitioners and researchers suffering from skewed data problems in their respective fields as we present the work to other domain-experts, especially those dealing with multiple minority classes. Since our ensemble methodology doesn’t overfit on the rare classes but tries to generalize on the non-major classes thus achieving a trade-off on overall accuracy but high robustness.

### Acknowledgements

We would like to thank Meenakshi Sarkar and Shivam Saboo for insightful discussions, also the anonymous reviewers for their valuable feedback on the draft. Authors would like to give a shout-out to Weights & Biases and to the ICLR’20 CCAI Workshop’s Mentorship Program.

PP extends special thanks to Debashish Ghose (IISc-B) and Krikamol Muandet (MPI-IS) for supporting this work.

### References


The solutions in the top (more accurate) - left (less variance) section of the Figure are most desirable i.e. accurate and robust.

We observe various models which were performing very well as per the vanilla validation-accuracy plummet when plotted w.r.t. balanced validation accuracy thus exposing the deep-rooted focus on the frequent class.
Figure 4: Sample images from our GAN training stages

(a) Initial: 7 epochs
(b) Midway: 20 epochs
(c) Final: 45 epochs
(d) Original Dataset c/o W&B Inc.(ILRI-Cornell-UCSD) [15]


[8] Mariano Focareta, Salvo Marcuccio, Silvia Liberata Ullo,


