ESTIMATING EXAMPLE DIFFICULTY USING VARIANCE OF GRADIENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

In machine learning, a question of great interest is understanding what examples are challenging for a model to classify. Identifying atypical examples helps inform safe deployment of models, isolates examples that require further human inspection, and provides interpretability into model behavior. In this work, we propose Variance of Gradients (VOG) as a valuable and efficient proxy metric for detecting outliers in the data distribution. We provide quantitative and qualitative support that VOG is a meaningful way to rank data by difficulty and to surface a tractable subset of the most challenging examples for human-in-the-loop auditing. Data points with high VOG scores are far more difficult for the model to learn and over-index on corrupted or memorized examples.

1 INTRODUCTION

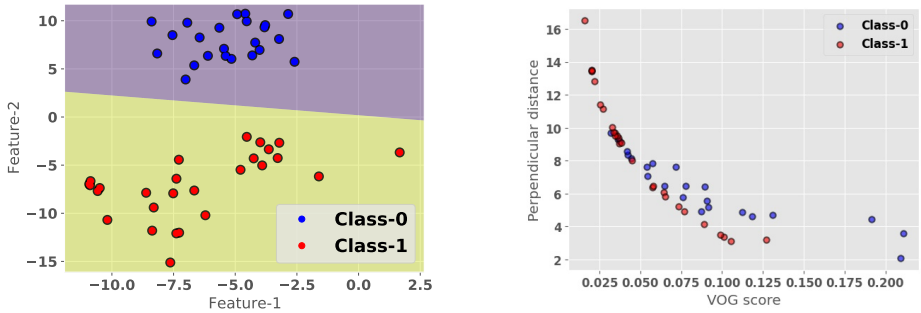
Reasoning about model behavior is often easier when presented with a subset of data points that are relatively more difficult for a trained model to learn. This not only aids interpretability through case based reasoning (Kim et al., 2016; Caruana, 2000; Hooker et al., 2019), but can also be used as a mechanism to surface a tractable subset of atypical examples for further human auditing (Leibig et al., 2017; Zhang, 1992; Hooker et al., 2019), for active learning to inform model improvements, or to choose not to classify certain examples when the model is uncertain (Bartlett & Wegkamp, 2008; Cortes et al., 2016).

One of the biggest bottlenecks for human auditing is the large scale size of modern datasets and the cost of annotating each feature (Veale & Binns, 2017). Methods which automatically surface a subset of relatively more challenging examples for human inspection help prioritize limited human annotation and auditing time. Despite the urgency of this use-case, ranking examples by difficulty has had limited treatment in the context of deep neural networks due to the computational cost of ranking a high dimensional feature space. Recent work in this direction has either been limited to small scale datasets or features a computational cost which is infeasible for most practitioners (Hooker et al., 2019; Carlini et al., 2019; Koh & Liang, 2017).

In this work, we start with a simple hypothesis – examples that a model has difficulty learning will exhibit higher variance in gradient updates over the course of training. On the other hand, we expect the backpropagated gradients of the samples that are *relatively easier* to learn will have lower variance because performance on that example does not consistently dominate the loss over the course of training. The gradient updates for the relatively easier examples are expected to stabilize early in training and converge to a narrow range of values. We term this class normalized ranking mechanism *Variance of Gradients* VOG, and demonstrate across a variety of large scale datasets that it efficiently ranks the difficulty of both training and test examples. VOG can be computed using either the predicted or true label, making it a valuable unsupervised auditing tool at test time when the true label is unknown.

Validating the behavior of VOG on artificial data. To begin, we illustrate the principle and effectiveness of VOG in a contrived toy example setting. The data was generated using two separate isotropic Gaussian clusters with a total of 500 data points. In such a simple low dimensional problem, the most challenging examples for the model to classify are closer to the decision boundary. In Fig. 1a we visualize the trained decision boundary of a multiple layer perceptron (MLP) with a single hidden layer trained for 15 epochs. VOG is computed at relative intervals for each training data

point. In Fig. 1b, we plot the final VOG score (Sec. 2) against the distance to the trained boundary. As expected, VOG allocates the highest scores to the most challenging examples that are closest to the decision boundary which exhibit the greatest variance in gradient updates over the course of the training process.



(a) Toy dataset trained decision boundary

(b) Distance vs. VOG score

Figure 1: We compute the variance of gradients (VOG) for each training data point in this two dimensional toy problem. On the right, we show that VOG accords higher scores to the most challenging examples closest to the decision boundary (as measured by the perpendicular distance).

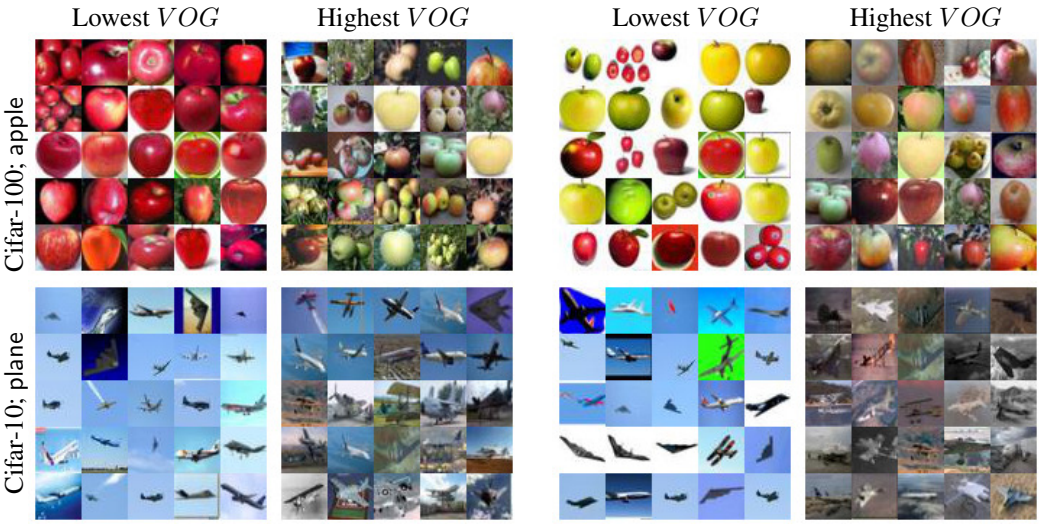
Contributions We scale this toy experiment and demonstrate consistent results across two different architectures and three datasets – Cifar-10, Cifar-100 Krizhevsky et al. (2009) and ImageNet (Russakovsky et al., 2015). Our contributions can be enumerated as follows:

1. **We propose Variance of Gradienst (VOG)** – a class-normalized variance gradient score for determining the relative ease of learning data samples within a given class (Sec. 2).
2. We show that VOG is an effective auditing tool for ranking high dimensional datasets by difficulty. VOG assigns higher scores to test-set examples that are more challenging for the model to classify. Restricting evaluation to the test-set examples with the lowest VOG greatly improves generalization performance. (Sec. 3).
3. VOG **identifies clusters of images with clearly distinct semantic properties.** As seen in Fig. 4), Low scores feature images with far less cluttered backgrounds and more prototypical vantage points of the object. In contrast, images with high scores over-index on images with cluttered backgrounds and atypical vantage points of the object of interest (zoomed in on part of the object, side profile of the object, shot from above).
4. VOG **effectively surfaces OOD and memorized examples** We empirically show that VOG allocates higher scores to examples that require *memorization* (Sec. 5) and out-of-distribution examples from curated benchmarks like ImageNet-O (Hendrycks et al., 2019a). We use VOG to explore how learning differs at different stages of training and show that VOG rankings are sensitive to the stage of training and provide insight into the learning process in deep neural networks.

Implications of this work It is becoming increasingly important for deep neural networks (DNNs) to make decisions that are interpretable to both researchers and end-users. In sensitive domains such as health care diagnostics (Xie et al., 2019; Gruetzemacher et al., 2018; Badgeley et al., 2019; Oakden-Rayner et al., 2019), self-driving cars (NHTSA, 2017) and hiring (Dastin, 2018; Harwell, 2019) providing tools for domain experts to audit models is of upmost importance. Our work offers an efficient method to rank the global difficulty of examples and automatically surface a possible subset to aid human interpretability. VOG can be computed using checkpoints stored over the course of training and is model agnostic. Critically, VOG can be computed using the predicted label, which makes it an unsupervised auditing tool at test time.

2 Methodology

We consider a supervised classification problem where a DNN is trained to approximate the function F that maps an input variable X to an output variable Y, formally $F : X \mapsto Y$. Without loss of generality we represent the image input as a feature vector $\mathbf{x}$. $y \in Y$ is a discrete label vector



(a) Early Training

(b) Late-stage Training

Figure 2: The 5×5 grid shows the top-25 Cifar-10 and Cifar-100 training-set images with the lowest and highest VOG scores in the *Early* (a) and *Late* (b) training stage respectively of two randomly chosen classes. Lower VOG images evidence uncluttered backgrounds (for both apple and plane) in the *Late* training stage. VOG also appears to capture a color bias present during the *Early* training stage for both apple (red). The VOG images in *Late* training stage present unusual vantage points, with images where the frame is zoomed in on the object of interest.

associated with each input x. Each label y corresponds to one of C categories or classes. A given input image $\mathbf{x}$ can be decomposed into a set of pixels $\{x_i\}_{i=1}^{N}$. For each image in the training and test set, we compute the gradient of the A_n^l with respect to each pixel x_i. Here, l designates the pre-softmax layer of the network and p is the index of either the true or the predicted class probability.

$$\mathbf{S} = \frac{\partial A_p^l}{\partial x_i}$$

This formulation may feel familiar as it is often computed based upon the weights of a trained model and visualized as a image heatmap for interpretability purposes (Baehrens et al., 2010; Simonyan et al., 2013). Here, we instead intend to compute the average variance of the input gradients for the same image across training to arrive at a scalar score that is a proxy measure of how challenging the example is to learn.

For $x_i \in x$, we compute the input gradients $\boldsymbol{S}_{ti}$ at different intervals across training, where t indicates the epoch number. We end up with a set of K gradient snapshots computed at different input steps. We then calculate the mean gradient input over all epochs in training μ_i:

$$\mu_i = \frac{1}{K}\sum_{t=1}^{K} \boldsymbol{S}_{ti} \tag{1}$$

For example, if the number of epochs is 300 and the interval between each snapshot is 10, $K = 30$. We note that VOG can be computed for both training and test sets by measuring the variance in gradients using checkpoints stored at different points in training. Finally, we compute VOG for each data sample:

$$VOG_i = \frac{1}{N}\sum_{i=1}^{N}\sqrt{\Big(\frac{1}{K}\sum_{t=1}^{K}\big(\boldsymbol{S}_{ti} - \mu_i\big)^2\Big)} \tag{2}$$

Here, N is the total number of pixels in a given image. Hence, for every data sample $x \in X$ we compute a scalar value indicating the variance in gradients score. In order to account for inherent differences in variance between classes, we normalize the absolute ranking of the VOG score by

class-level mean and standard deviation. This amounts to asking: *What is the variance of gradients for this image relative to all other exemplars for this class category?*

2.1 Experimental Setup

Datasets: We evaluate our methodology on Cifar-10 and Cifar-100 Krizhevsky et al. (2009) and ImageNet (Russakovsky et al., 2015).

Cifar Training: We use a ResNet-18 network (He et al., 2016) for both Cifar-10 and Cifar-100. For each dataset, we train for 350 epochs using stochastic gradient descent (SGD) and compute the input gradients for each sample every 10 epochs. We implemented standard data augmentation by applying cropping and horizontal flips of input images. We use a base learning rate schedule of 0.1 and adaptively change to 0.01 at 150th and 0.001 at 250th training epochs. The top-1 test-set accuracy for Cifar-10 and Cifar-100 were 89.57% and 66.86% respectively.

ImageNet Training: We use a ResNet-50 (He et al., 2015) trained on ImageNet. The network was trained with batch normalization (Ioffe & Szegedy, 2015), weight decay, decreasing learning rate schedules, and augmented training data. We train for $32,000$ steps (approximately 90 epochs) on ImageNet with a batch size of 1024 images. We store 32 checkpoints over the course of training, but in practice observe that VOG ranking is very stable computed with as few as 3 checkpoints. Our model achieves a top-1 accuracy of 76.68% and top-5 accuracy of 93.29%.

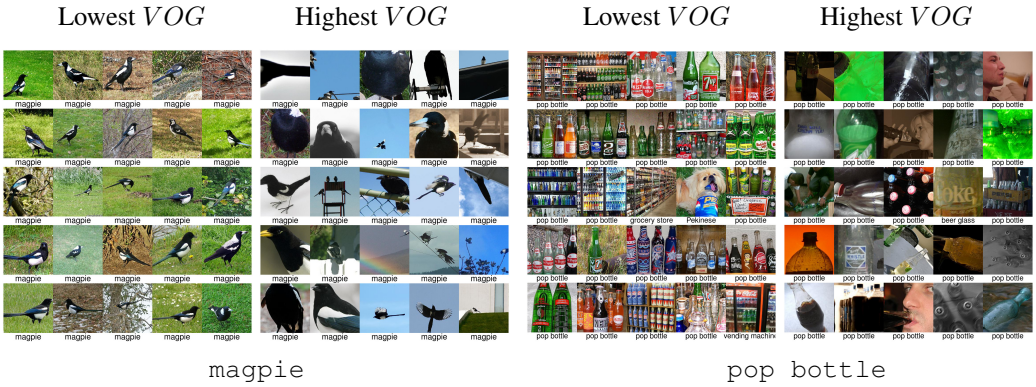


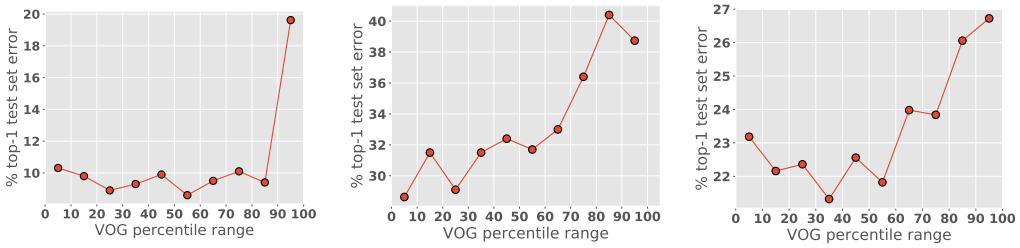
Figure 3: Each 5×5 grid shows the top-25 ImageNet training-set images with the lowest and highest VOG scores for the class `magpie` and `pop bottle` with their predicted labels below the image. Training set images with higher VOG scores (b) tend to feature zoomed-in images with atypical color schemes and vantage points.

3 Evaluating VOG

For all datasets considered, we compute VOG for both training and eval sets. In this section, we evaluate the utility of VOG as an auditing tool. We evaluate the stability of the VOG ranking, measure how discriminative it is at seperating easy examples from difficult, and comment on the qualitative properties of images at either end of the VOG spectrum.

Qualitative inspection of ranking A qualitative inspection of examples with high and low VOG scores shows that there are distinct properties to the images at either end of the ranking. We visualize 25 images ranked lowest and highest according to VOG for both the entire dataset (visualized for ImageNet in Fig. 5) and for specific classes (visualized for ImageNet in Fig. 3 and for Cifar-10 and Cifar-100 in Fig. 2). We observe that ranking by VOG produces clusters with clearly distinguished semantic features. Images with low VOG score tend to have uncluttered and often white backgrounds with the object of interest centered clearly in the frame. Images with the *highest* VOG scores have cluttered backgrounds and the object of interest is not easily distinguishable from the background. We also note that images with high VOG score tend to feature atypical vantage points of the objects such as highly zoomed frames, side profiles of the object or shots taken from above. Other, the object of interest is partially occluded or there are image corruptions present such as heavy blur.

Test-set error and VOG A valuable property of an auditing tool is to be able to effectively discriminate between easy and challenging examples. Here, we measure whether VOG is able to do so. In Fig. 4, we plot the test-set error of examples bucketed by VOG decile. For this and the remainder of the experiments, we compute VOG using checkpoints stored from the first and last 3 epochs. Thus, at each point of the x-axis, we are computing the test-set error on the 10% of data whose VOG score falls between each decile. Note that we plot error, so lower is better. We show that examples at the lowest percentiles of VOG have far lower error rates. Mis-classification increases with an increase in VOG scores. Our results are consistent across all datasets, yet more pronounced for the more complex datasets Cifar-100 and ImageNet. We ascribe this to differences in underlying model complexity. In Fig. 10 we observe that test-set accuracy on the lowest VOG scored images improves beyond baseline test-set performance. Our models show improved generalization when restricted to low VOG images.



(a) Cifar-10 (b) Cifar-100 (c) ImageNet

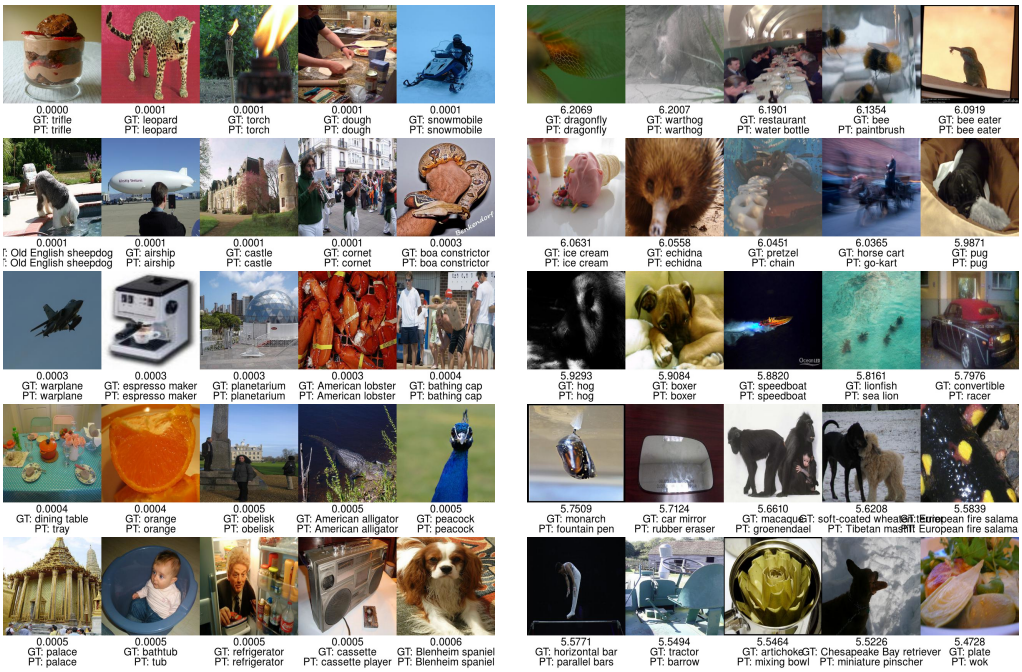
Figure 4: The mean top-1 test set error (y-axis) for the exemplars thresholded by VOG score percentile (x-axis). Across Cifar-10, Cifar-100 and ImageNet, we observe that misclassification increases with an increase in VOG scores. Across all datasets we observe that the group of samples in the top-10 percentile VOG scores have the highest error rate, *i.e.*, contains most number of misclassified samples. For all datasets, model generalization improves on the bottom 10th percentile relative to the entire dataset.

VOG as an unsupervised auditing tool Many auditing tools to evaluate and understand possible model bias require the presence of labels for protected attributes and underlying variables. However, this is highly infeasible in real-world settings (Veale & Binns, 2017). For image and language datasets, the high dimensionality of the problem makes it hard to identify a priori what are underlying variables to be aware of. Even acquiring the labels for a limited number of attributes protected by law (gender, race) is expensive and/or may be perceived as intrusive leading to noisy or incomplete labels. One key advantage of VOG is that we show it continues to produce a useful ranking even when the gradients are computed with respect to the predicted label. In Fig. 5, we include the top and bottom 25 VOG ImageNet test images using predictions. We also computed the mean test-error for the predicted VOG distribution, and find that it also effectively discriminates between top-10 and bottom-10 examples with 73.6% and 77.5% accuracy respectively (Fig. 11).

Stability of VOG ranking To build trust with the end-user, a key desirable property of any auditing tool is consistency in performance. We would expect a consistent method to produce a ranking with a closely bounded distribution of scores across independently trained runs of the same model architecture and dataset. To measure the consistency of the VOG ranking, we train 5 Cifar-10 networks from random initialization using the training methodology explained in Sec. 2.1. Empirically, Fig. 6a shows that VOG rankings evidence a consistent distribution of test-error at each percentile given the same model and dataset.

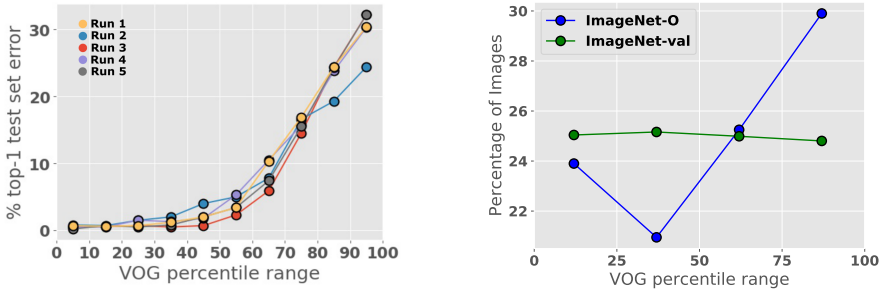
4 Leveraging VOG to understand early and late training dynamics

Recent work has shown that there are distinct stages to training in deep neural networks (Achille et al., 2017; Jiang et al., 2020; Mangalam & Prabhu, 2019; Faghri et al., 2020). In our second set of experiments, we explore whether rankings according to VOG are sensitive to the stage of the training process. Hence, we compute VOG separately for two different stages of the training process,



(a) Lowest VOG (b) Highest VOG

Figure 5: Each 5×5 grid shows the top-25 ImageNet test-set images with the lowest and highest VOG scores for the top-1 predicted class. Test set images with higher VOG scores tend to feature zoomed-in images and are misclassified more as compared to the lower VOG images which tend to feature more prototypical vantage points of objects.



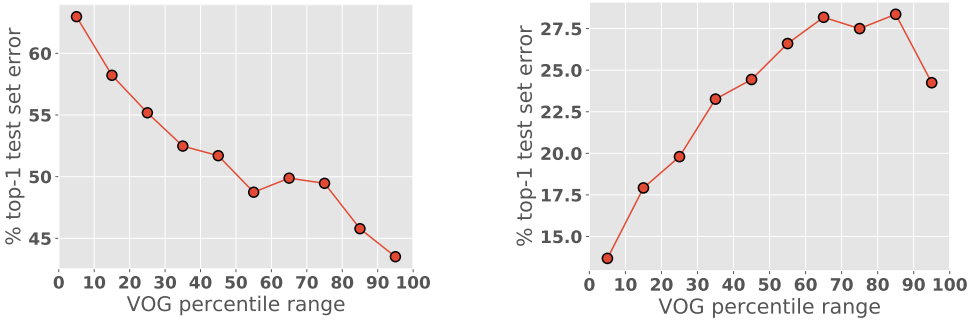
(a) (b)

Figure 6: **Left:** Consistency in ranking is an important attribute of any auditing tool. Here, we plot the VOG top-1 test set error for 5 resnet-18 networks independently trained on Cifar-10 from random intialization. The plots show that VOG produces a stable ranking with a similar distribution of error in each percentile across all images. **Right:** We measure the distribution of ImageNet-O images across percentiles. We find that higher percentiles of VOG over-index on these out of distribution images.

which we term (1) the *Early* stage (first three epochs), and (2) the *Late* stage (last three epochs). Test-set accuracy at the *Early* stage is 44.65%, 14.16% and 51.87% for Cifar-10, Cifar-100 and ImageNet respectively. In the *Late* stage it is 89.57%, 66.86% and 76.69% for Cifar-10, Cifar-100 and ImageNet respectively.

We find that there is a noticeable visual difference between the image ranking computed for *Early* and *Late* stages of training. As seen in Fig. 2, for some classes such as `apple` it appears that VOG scores also capture network color bias present during the *Early* training stages. For these classes, the lowest VOG scores over-index on red colored apples.

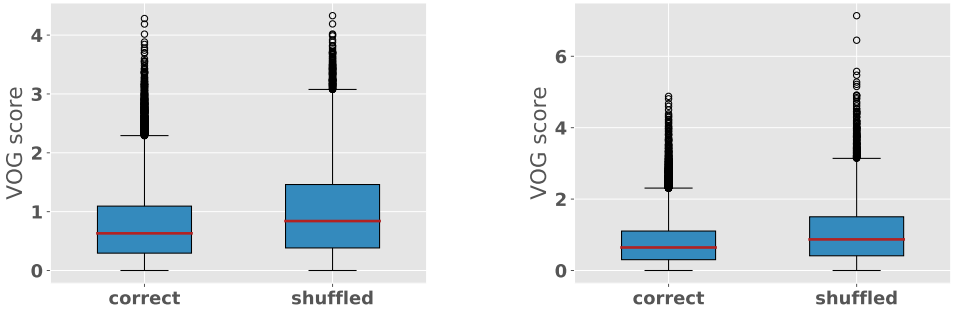
For ImageNet, we compute the distribution of error in early and late stage and find a remarkable flipping point (Fig. 7). During the early training stage, samples having higher VOG score tend to have a lower error rate as the gradient updates center on easy examples. This phenomenon is reversed during the late stage of the training where most easy example have been learnt, and updates to the harder examples dominate the the computation of variance. Hence, samples having high VOG results in higher error rate.



(a) Early stage

(b) Late stage

Figure 7: The mean top-1 test set error (y-axis) for the exemplars thresholded by VOG score percentile (x-axis) in ImageNet validation set. The Early (a) and Late (b) stage VOG analysis shows inverse behavior where the role of VOG flips as the training progresses.



(a) Cifar-10

(b) Cifar-100

Figure 8: Box-plot of subset the VOG distribution of all examples with *correct* labels against the 20% of the dataset with *shuffled* labels. It is visible that the distribution of VOG scores, both the mean (red line in the plot) and spread, for shuffled data is higher than that of the correct samples for both Cifar-10 (right plot) and Cifar-100 (left plot).

5 Relationship between VOG Scores and Memorized/OOD Examples

Recent work has highlighted that deep neural networks produce output probabilities that are uncalibrated (Guo et al., 2017; Kendall & Gal, 2017; Lakshminarayanan et al., 2017; Hendrycks & Gimpel, 2016) and thus cannot be interpreted as a measure of certainty. If VOG is a useful auditing tool, we expect it to capture model uncertainty even when this is not reflected in the end probabilities.

To this end, we consider VOG rankings on two tasks where the network produces highly certain predictions for incorrect or out-of-distribution inputs.

Surfacing examples that require memorization Overparameterized networks have been shown to achieve zero training error by memorizing examples (Zhang et al., 2016; Feldman, 2020). We explore whether VOG is able to identify examples that require memorization and the rest of the dataset. To do this, we replicate the general experiment setup of Zhang et al. (2016) and replace 20% of all labels

in the training set with random shuffled labels. We re-train the model from random initialization and compute VOG scores at relative intervals *across training* for all examples in the training set. Our network achieves 0% training error which would only be possible given successful memorization of the noisy examples with shuffled labels. *Is VOG able to discriminate between these memorized examples and the rest of the dataset?* In Fig.8 we plot the box plot distribution of VOG scores for the subset of the data with *shuffled* labels that required memorization beside *correct* labels. We find that the mean and spread of the examples with the shuffled labels are higher when compared to the rest of the dataset.

ImageNet-O experiments We consider ImaageNet-O (Hendrycks et al., 2019b), an open source curated out-of-distribution (OOD) dataset designed to fool classifiers. ImageNet-O consists of images that are not included in the original 1000 ImageNet classes. These images were selected with the goal of producing high confidence incorrect ImageNet-1K predictions of labels from within the training distribution. We are interesting in understanding if VOG can correctly rank Imagenet-O examples as being atypical or out of distribution. We would expect to observe that ImageNet-O examples would be overrepresented in top percentiles of VOG scores vs the lowest scores. In Fig. 6b, we see that this is indeed the case. We plot the count of ImageNet-O image in each percentile as a fraction of the total count of ImageNet-O images (2000 images in total). ImageNet-O images are relatively over-represented at high levels of VOG, with 30% of all images in the top-25th percentile vs 24% in the bottom 25th percentile.

6 Related Work

Our work proposes a method to rank training and test examples by estimated difficulty. Given the size of modern day datasets (Hooker, 2020), this can be a powerful interpretability tool to isolate a tractable subset of examples for human-in-the-loop audting and also aid in curriculum learning (Bengio et al., 2009). Prior work has proposed different notions of what subset merits surfacing. Early work by (Zhang, 1992; Bien & Tibshirani, 2012; Kim et al., 2015; Kim et al., 2016) that introduced the notion of prototypes, quintessential examples in the dataset, but did not focus on deep neural networks. Kim et al. (2016) also requires assumptions about the statistics of the input distribution. Gal & Ghahramani (2016) showed how we can use dropouts as a bayesian approximation method for representing model uncertainty in deep learning. Work by Li et al. (2017) requires modifying the architecture to prefix an autoencoder in order to surface a set of prototypes. Koh & Liang (2017) proposes influence functions to identify training points most influential on a given prediction.

Unlike previous works, we propose a measure that can be extended to rank the entire dataset by estimated difficulty (rather than surfacing a prototypical subset). Additionally, ranking individual samples using methods like Koh & Liang (2017) would be extremely computationally extensive. Our method does not require modifying the architecture or making any any assumptions about the statistics of the input distribution. In that sense, our work is complementary to recent work by Jiang et al. (2020) which proposes a c-score to rank each example by alignment with the training instances, Hooker et al. (2019) which classify examples as outliers according to sensitivity to varying model capacity and Carlini et al. (2019) which consider several different measures to isolate prototypes that could conceivably be extended to rank the entire dataset. We note that the c-score method proposed by Jiang et al. (2020) is considerably computationally intensive to compute than VOG as it requires training up to $20,000$ network replications per data set. Several of the propotype methods considered by Carlini et al. (2019) require training ensembles of models, as does the compression sensitivity measure proposed by Hooker et al. (2019). Our method is both different in formulation and can be leveraged using a small number of existing checkpoints saved over the course of training.

7 Conclusion and Future Work

Our methodology offers one way for humans to better understand the relative difficulty of different examples. One of our key findings is that VOG is far more challenging to classify for the algorithm and surfaces clusters of images with distinct visual properties. VOG is straight-forward to compute and can take advantage of current best practices of storing multiple checkpoints over the course of training. In practice, a domain expert may choose to compute VOG for a class of particular interest which would further reduce the computational cost.

REFERENCES

Alessandro Achille, Matteo Rovere, and Stefano Soatto. Critical learning periods in deep neural networks. *ArXiv*, abs/1711.08856, 2017.

Marcus Badgeley, John Zech, Luke Oakden-Rayner, Benjamin Glicksberg, Manway Liu, William Gale, Michael McConnell, Bethany Percha, and Thomas Snyder. Deep learning predicts hip fracture using confounding patient and healthcare variables. *npj Digital Medicine*, 2:31, 04 2019. doi: 10.1038/s41746-019-0105-1.

David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert MÃžller. How to explain individual classification decisions. *Journal of Machine Learning Research*, 11(Jun):1803–1831, 2010.

Peter L. Bartlett and Marten H. Wegkamp. Classification with a reject option using a hinge loss. *J. Mach. Learn. Res.*, 9:1823–1840, June 2008. ISSN 1532-4435. URL http://dl.acm.org/citation.cfm?id=1390681.1442792.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pp. 41–48, 2009.

Jacob Bien and Robert Tibshirani. Prototype selection for interpretable classification. *arXiv e-prints*, art. arXiv:1202.5933, February 2012.

Nicholas Carlini, Úlfar Erlingsson, and Nicolas Papernot. Distribution Density, Tails, and Outliers in Machine Learning: Metrics and Applications. *arXiv e-prints*, art. arXiv:1910.13427, October 2019.

Rich Caruana. Case-based explanation for artificial neural nets. In Helge Malmgren, Magnus Borga, and Lars Niklasson (eds.), *Artificial Neural Networks in Medicine and Biology*, pp. 303–308, London, 2000. Springer London. ISBN 978-1-4471-0513-8.

Corinna Cortes, Giulia DeSalvo, and Mehryar Mohri. Boosting with abstention. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 29*, pp. 1660–1668. Curran Associates, Inc., 2016. URL http://papers.nips.cc/paper/6336-boosting-with-abstention.pdf.

Jeffrey Dastin. Amazon scraps secret ai recruiting tool that showed bias against women. *Reuters*, 2018. URL https://reut.rs/2p0ZWqe.

Fartash Faghri, David Duvenaud, David J. Fleet, and Jimmy Ba. A Study of Gradient Variance in Deep Learning. *arXiv e-prints*, art. arXiv:2007.04532, July 2020.

Vitaly Feldman. Does learning require memorization? a short tale about a long tail. *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, 2020.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pp. 1050–1059, 2016.

Ross Gruetzemacher, Ashish Gupta, and David B. Paradice. 3d deep learning for detecting pulmonary nodules in ct scans. *Journal of the American Medical Informatics Association : JAMIA*, 25 10: 1301–1310, 2018.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On Calibration of Modern Neural Networks. *arXiv e-prints*, art. arXiv:1706.04599, Jun 2017.

Drew Harwell. A face-scanning algorithm increasingly decides whether you deserve the job. *The Washington Post*, 2019. URL https://wapo.st/2X3bupO.

K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. *ArXiv e-prints*, December 2015.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Dan Hendrycks and Kevin Gimpel. A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks. *arXiv e-prints*, art. arXiv:1610.02136, Oct 2016.

Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. *CoRR*, abs/1907.07174, 2019a. URL `http://arxiv.org/abs/1907.07174`.

Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. *arXiv preprint arXiv:1907.07174*, 2019b.

Sara Hooker. The hardware lottery, 2020.

Sara Hooker, Aaron Courville, Gregory Clark, Yann Dauphin, and Andrea Frome. What Do Compressed Deep Neural Networks Forget? *arXiv e-prints*, art. arXiv:1911.05248, November 2019.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

Ziheng Jiang, Chiyuan Zhang, Kunal Talwar, and Michael C. Mozer. Characterizing Structural Regularities of Labeled Data in Overparameterized Models. *arXiv e-prints*, art. arXiv:2002.03206, February 2020.

Ziheng Jiang, Chiyuan Zhang, Kunal Talwar, and Michael C Mozer. Exploring the memorization-generalization continuum in deep learning. *arXiv preprint arXiv:2002.03206*, 2020.

Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 30*, pp. 5574–5584. Curran Associates, Inc., 2017.

Been Kim, Cynthia Rudin, and Julie Shah. The Bayesian Case Model: A Generative Approach for Case-Based Reasoning and Prototype Classification. *arXiv e-prints*, art. arXiv:1503.01161, March 2015.

Been Kim, Rajiv Khanna, and Oluwasanmi O Koyejo. Examples are not enough, learn to criticize! criticism for interpretability. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 29*, pp. 2280–2288. Curran Associates, Inc., 2016.

Pang Wei Koh and Percy Liang. Understanding Black-box Predictions via Influence Functions. *arXiv e-prints*, art. arXiv:1703.04730, March 2017.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 6405–6416, USA, 2017. Curran Associates Inc. ISBN 978-1-5108-6096-4. URL `http://dl.acm.org/citation.cfm?id=3295222.3295387`.

Christian Leibig, Vaneeda Allken, Murat Seckin Ayhan, Philipp Berens, and Siegfried Wahl. Leveraging uncertainty information from deep neural networks for disease detection. *Scientific Reports*, 7, 12 2017. doi: 10.1038/s41598-017-17876-z.

Oscar Li, Hao Liu, Chaofan Chen, and Cynthia Rudin. Deep learning for case-based reasoning through prototypes: A neural network that explains its predictions. *CoRR*, abs/1710.04806, 2017. URL `http://arxiv.org/abs/1710.04806`.

Karttikeya Mangalam and Vinay Uday Prabhu. Do deep neural networks learn shallow learnable examples first. 2019.

NHTSA. Technical report, U.S. Department of Transportation, National Highway Traffic, Tesla Crash Preliminary Evaluation Report Safety Administration. *PE 16-007*, Jan 2017.

Luke Oakden-Rayner, Jared Dunnmon, Gustavo Carneiro, and Christopher Ré. Hidden Stratification Causes Clinically Meaningful Failures in Machine Learning for Medical Imaging. *arXiv e-prints*, art. arXiv:1909.12475, Sep 2019.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, Dec 2015. ISSN 1573-1405. doi: 10.1007/s11263-015-0816-y. URL https://doi.org/10.1007/s11263-015-0816-y.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

Michael Veale and Reuben Binns. Fairer machine learning in the real world: Mitigating discrimination without collecting sensitive data. *Big Data & Society*, 4(2):2053951717743530, 2017. doi: 10.1177/2053951717743530. URL https://doi.org/10.1177/2053951717743530.

Hongtao Xie, Dongbao Yang, Nannan Sun, Zhineng Chen, and Yongdong Zhang. Automated pulmonary nodule detection in ct images using deep convolutional neural networks. *Pattern Recognition*, 85:109 – 119, 2019. ISSN 0031-3203. doi: https://doi.org/10.1016/j.patcog.2018.07.031. URL http://www.sciencedirect.com/science/article/pii/S0031320318302711.

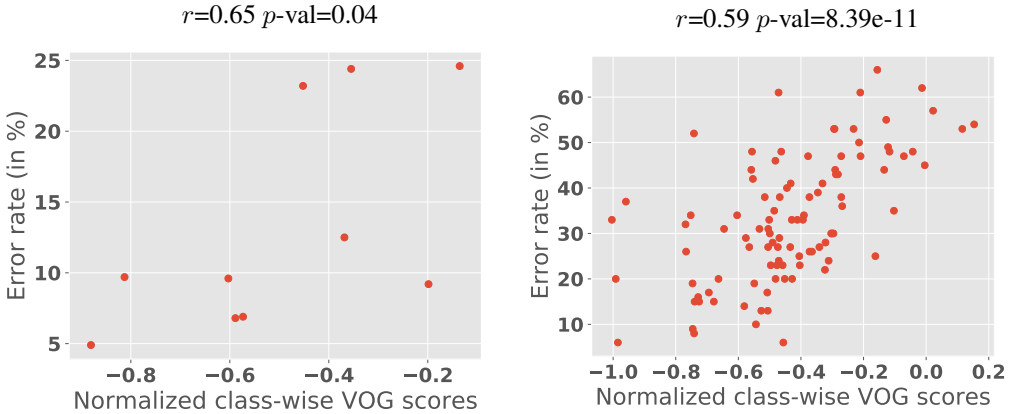
Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.

Jianping Zhang. Selecting typical instances in instance-based learning. In Derek Sleeman and Peter Edwards (eds.), *Machine Learning Proceedings 1992*, pp. 470 – 479. Morgan Kaufmann, San Francisco (CA), 1992. ISBN 978-1-55860-247-2. doi: https://doi.org/10.1016/B978-1-55860-247-2.50066-8. URL http://www.sciencedirect.com/science/article/pii/B9781558602472500668.

A APPENDIX

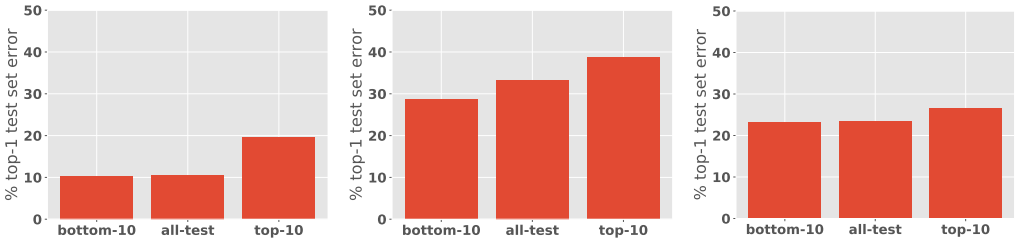
Toy Experiment We generate the clusters for classification using scikit-learn library. We used a 90-10% split for dividing the dataset into the training and testing category. A linear Multiple Layer Perceptron (MLP) network with a single hidden layer of 10 neurons was trained using Stochastic Gradient Descent (SGD) optimizer for 15 epochs. We divided the training process into three stages: (1) $Early$ [0, 5) epochs, (2) $Middle$ [5, 10), and (3) $Late$ stage [10, 15). Our trained model achieves a 0% testing error using a linear boundary (Fig. 1a).

Class Level Error Metrics and VOG Here, we explore whether VOG is able to capture class level differences in difficulty. We compute VOG scores for each image in the test-set of Cifar-10 and Cifar-100 (both test-sets have $10,000$ images). In Fig. 9, we plot the average absolute VOG score for each class against the false negative rate for each class. We find that there is a positive, albeit weak, correlation between the two, classes with higher VOG scores have higher mis-classification error rate. The correlation between these metrics is 0.65 and 0.59 for Cifar-10 and Cifar-100 respectively. Given that VOG is computed a per-example level, we find it interesting that the aggregate average of VOG is able to capture class level differences in difficulty.



(a) Cifar-10 (b) Cifar-100

Figure 9: Plot of class false negative rate (y-axis) against average class VOG score for all classes (x-axis). **Left**: Cifar-10 **Right**: Cifar-100. There is a statistically significant positive correlation between class level error metrics and average VOG score (alpha set at 0.05).



(a) Cifar-10 (b) Cifar-100 (c) ImageNet

Figure 10: Bar plots showing the mean top-1 error rate (in $\%$) for three group of samples from (1) the subset of the test-set with the bottom 10th percentile of VOG scores, (2) the complete testing dataset, and (3) the subset of the test-set with the top 10th percentile of VOG scores.

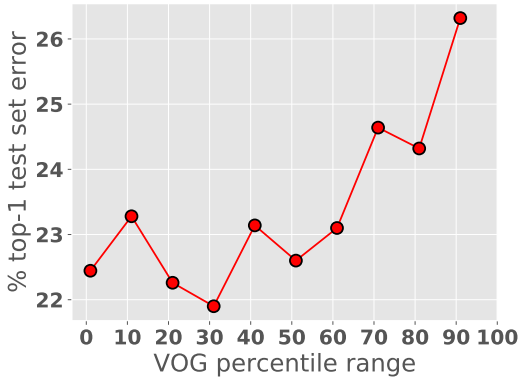


Figure 11: The mean top-1 test set error (y-axis) for the exemplars thresholded by VOG score percentile (x-axis) calculated using the predicted labels. We observe that misclassification increases with an increase in VOG scores. Across ImageNet we observe that VOG calculated for the predicted labels follows the general trend as in Fig. 5 where the top-10 percentile VOG scores have the highest error rate.