CS260D Project Report - Examining Performance of Training Data Subsets Selected by SAS when the Proxy Model is Trained on a Different Dataset

Matt Abuzalaf [*1] **Ryan Dunker** [*1] **Yeonje Jo** [*1] **Lauren Mizner** [*1]

Abstract

In this project, we investigated the usability of the SAS algorithm in selecting high-quality training subsets for a given dataset when the proxy model has been pre-trained on a different dataset. For our experiments, we used a model pre-trained on CIFAR100 data to perform SAS subset selection for six datasets with diverse classification tasks and numbers of labels (CIFAR100, Food101, CIFAR10, SVHN, GTSRB, and DTD), and evaluated the predictive abilities of these subsets compared to randomly selected subsets and full data in a downstream supervised classification task. Results indicated that SAS subsets predictably outperformed random for CIFAR100 but underperformed for CIFAR10, SVHN, GTSRB, and DTD, which each contained far fewer classes than CIFAR100. Results indicate that SAS subsets may outperform random for Food101, which contains a similar number of classes to CIFAR100.

1. Introduction

Since effectively training a deep neural network model is a time- and resource-intensive process, particularly for models with large numbers of parameters that are the current state of the art for challenging tasks like image classification, areas of research focus have emerged examining how to make the training process more efficient. One such focus is subset selection, in which the model is trained on only a subset of the available training data with the goal of achieving similar predictive accuracy to a model trained on the full training set. For a subset that effectively achieves this goal, the resulting speedup of the training process would be proportional to the degree of dataset size reduction (e.g., training on a subset containing 25% of the data could result in roughly 4x speedup).

Of interest in this project is the Subsets that maximize Augmentation Similarity (SAS) algorithm, a pre-training procedure that uses contrastive self-supervised learning (SSL) on training image data points to identify examples that can comprise an effective subset. To reflect the idea that high-quality labels may be expensive to attain for large datasets, SAS was configured to perform subset selection largely without access to training labels. A small percentage of labels (e.g., 1%), are made available to facilitate assigning every training data point to a latent class, and the data points returned by SAS are those which can act as "class centers" by best maximizing an image augmentation similarity function. This similarity function is governed by a proxy model, which helps to estimate expected augmentation distances and can be pretrained for the dataset of interest.

One novel use case for SAS could be using a proxy model that has been pre-trained on one dataset for performing subset selection on a new, similar dataset that does not have full labels available. In our project, we attempt to examine whether and how well using a proxy model pretrained for one dataset can transfer to subset selection for different datasets by conducting a wide-ranging ablation study, using a proxy model pre-trained on CIFAR100 to select subsets with SAS for six different datasets.

2. Related Work

2.1. CLIP (Contrastive Language-Image Pre-Training)

CLIP is a neural network trained on a variety of image and text pairs that is found to match the performance of ResNet50 on ImageNet "zero-shot" without using any of the original labeled examples (Radford et al., 2021). For our project, we will be utilizing the CLIP image joint encoder with labels available for 1% of the training data in order to assign each data point in the full training set to a latent class for the purposes of subset selection.

[1]University of California, Los Angeles, California, USA. Correspondence to: Matt Abuzalaf <606097658, mabuzalaf1996@g.ucla.edu>, Ryan Dunker <205329251, rydunker@g.ucla.edu>, Yeonje Jo <404332471, yeonjejo510@gmail.com>, Lauren Mizner <005225768, lmizner@g.ucla.edu>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

2.2. SAS (Subsets that maximize the expected Augmentation Similarity)

As mentioned, SAS is a method which finds subsets that minimize expected augmentation distance, or equivalently, that maximize expected augmentation similarity, by approximately finding expected augmentation distances for training images in a self-supervised fashion. The paper proposing SAS (Joshi & Mirzasoleiman, 2023) is the primary resource used for this project, as we aimed to follow its methods as closely as possible and used the linked codebase to assemble SAS and random subsets (though we used our own, smaller classifier model for the downstream classification task).

3. Problem Formulation

The goal of the project was to examine the limits of the transferability and applicability of a pre-trained CIFAR100 proxy model by using the model for SAS subset selection on datasets that are of differing similarities to CIFAR100 in terms of the involved classification tasks and numbers of classes. The approach for collecting and evaluating SAS results for a given dataset was already well-formulated in (Joshi & Mirzasoleiman, 2023), so in order to approach this task methodically, we aimed to follow the same workflow defined in the paper, which outlined upstream and downstream tasks. In the upstream task, SAS subsets of user-determined sizes are selected by assigning latent classes to each point in the training set, then selecting points to return by using unlabeled contrastive SSL aided by a proxy model. In the downstream task, the predictive ability of SAS-selected subsets are evaluated by training them (with labels available) on a classifier model, then comparing their accuracy to models trained on random subsets of the same size as well a model trained on full data. Our project will apply this process on six different datasets, using a pre-trained CIFAR100 proxy model on each, as outlined in Section 4.

4. Method

4.1. Dataset

For this project, we tested six RGB image classification datasets from PyTorch's list of built-in datasets[1] to examine the usability of a pre-trained CIFAR100 proxy model for performing SAS subset selection on non-CIFAR100 data. As shown in Table 1, the following datasets are used: Food101 (Bossard et al., 2014), CIFAR100 (Krizhevsky et al.), Describable Textures Dataset (DTD) (Cimpoi et al., 2014), German Traffic Sign Recognition Benchmark (GTSRB) (Houben et al., 2013), CIFAR10 (Krizhevsky), and Street View House Numbers (SVHN) (Yuval Netzer). CIFAR100 and CIFAR10 both involve classifying images respectively belonging to 100 and 10 animal or object labels. Food101 involves classifying images to 101 different food labels. DTD involves classifying "physical textures" displayed in images according to 47 labels such as "bumpy" or "wrinkled". GTSRB involves classifying images of different road signs to 43 labels. Finally, SVHN involves classifying images of address numbers on houses (cropped to show a single digit) to labels of 0 through 9.

We aimed in selecting these datasets to reflect (1) a broad variety of image classification tasks with (2) varying numbers of class labels, including two datasets with approximately100 labels, two with approximately 50, and two with 10. Since our model was relatively small to be trainable under system constraints, we did not include larger 200+ class prediction tasks with the consideration that this might not achieve meaningful results. All image data points were read in as resized 3x32x32 tensors to work with the SAS code and to standardize for our downstream classifier model.

Dataset	Number of classes
Food101	101
CIFAR100	100
DTD	47
GTSRB	43
CIFAR10	10
SVHN	10

Table 1. Summary of datasets

4.2. Subset selection

We proceeded with the same process used in the original SAS paper to select, test, and evaluate subsets. Namely, for each of the six selected datasets, we (1) used the SAS algorithm (with the pre-trained CIFAR100 proxy model) to select subsets of varying sizes using contrastive SSL, (2) randomly selected subsets of those same sizes, and (3) retained the full training dataset. Downstream classifier models would later be trained and evaluated for each subset.

For selecting subsets using SAS, the full training set of each dataset was run through the CLIP encoder (the "clip_approx" function in the SAS codebase) with 1% of training labels made available at random to assign a latent class to each training point. Then, the "SASSubsetDataset" function was used, provided with the CIFAR100 proxy model (a pre-trained ResNet architecture included in the codebase), to perform contrastive SSL and return a user-determined fraction of the training points as the SAS subset.

In the SAS paper, the experimental results indicated that some datasets (such as CIFAR10) approached the classification accuracy of the full training dataset when only a small subset of training points were available, whereas other

[1] https://pytorch.org/vision/stable/datasets.html

datasets (such as CIFAR100) required larger subset sizes such as 80%. Since we wanted to avoid making assumptions when using the CIFAR100 proxy model for SAS selection on different datasets, we opted to select eight different subset sizes for each of the six datasets - 5%, 10%, 15%, 20%, 30%, 40%, 60%, and 80% - so we could benchmark the relative performances of SAS and randomly selected subsets at a variety of points.

4.3. Model training

Continuing with the same general procedure outlined in the SAS paper, for the downstream classification task that evaluates the classification accuracy of subsets with full label data available, we trained a model architecture on each subset selected by SAS or randomly, as well as on the full training data, for each of the six datasets. With eight SAS subsets of varying size, eight random subsets, and full training data per dataset, this resulted in a total of 102 models that needed to be trained.

We made a handful of assumptions during training. Specifically, for each of the six datasets, the same model design and hyperparameter choices (including number of epochs, batch size, optimizers) should be used for all subsets and the full training data to maintain the fidelity of the results of model evaluation. For simplicity under the time constraints of the project, we also decided to implement one model design that could achieve reasonable results on all six datasets.

Building from the baseline of the simple CNN model included in the CIFAR10 Pytorch tutorial[2], we ultimately opted for a somewhat larger model design as described in Section 5. This model was still somewhat small compared to architectures that have achieved the highest classification accuracy on these datasets (and the ResNets used for training in the original SAS paper).

However, as our team was under time and CPU constraints (with some models on larger subset sizes requiring over an hour to train), we believe we generally achieved acceptable enough results for this exercise on the six datasets, with several datasets achieving high test accuracy when trained on full data and those with lower overall accuracy still managing to reach above-zero test accuracy on most if not all classes. We anticipate that with GPU resources, a larger model, and time to train for additional epochs as needed, higher fidelity results could have been achieved.

4.4. Performance metrics

The primary metrics we collected from our experiments included: top-1 accuracy (label corresponding to highest score in output layer is selected as the predicted class); number of classes with accuracy above zero (assumption is that we might have some zero-accuracies for datasets with larger numbers of classes); and per-class accuracies including worst-class and best-class accuracy.

5. Experiments

5.1. Upstream contrastive learning

As mentioned in Section 4, for obtaining SAS and random subsets for each of the six training datasets, 8 different sizes were selected: 5%, 10%, 15%, 20%, 30%, 40%, 60% and 80%. For SAS, we followed the example code procedure provided on the GitHub page for the SAS paper[3]. Latent class assignment was performed using CLIP with 1% of random training labels, and SAS subsets were obtained with the "SASSubsetDataset" function, using the provided ResNet proxy model pretrained for CIFAR100. For random selection: "RandomSubsetDataset" was used and provided the same fraction sizes as for SAS.

5.2. Downstream supervised learning

We experimentally determined the following CNN architeture would perform reasonably well for training on the full train data for each of the six datasets:

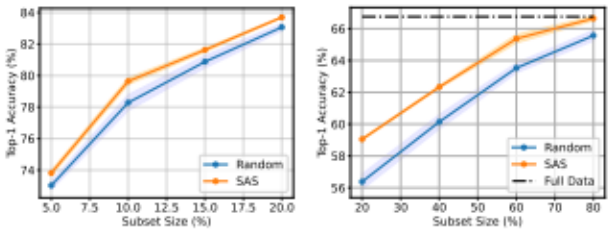
- Conv2d(3,32,3) $\rightarrow$ ReLU $\rightarrow$ MaxPool2d(2,2)
- Conv2d(32,64,3) $\rightarrow$ ReLU $\rightarrow$ MaxPool2d(2,2)
- Conv2d(64, 128,3) $\rightarrow$ ReLU $\rightarrow$ MaxPool2d(2,2)
- Linear(512, 256)
- Linear(256, 128)
- Linear(128, num_classes)

The model took 3-channel RGB images as input and consisted of 3 convolutional layers and 3 fully connected layers, with the output layer size corresponding to the number of classes for the dataset in question. Adam was experimentally determined to be the best performing optimizer (with default learning rate of 1e-3, β_1 of 0.9, and β_2 of 0.999), and cross-entropy loss was used for weight updates.

Under CPU constraints, we limited the batch size for training to 4 and number of workers to 2. For each model, we experimentally determined a "reasonable" number of training epochs that would reliably train the model to apparent convergence when using the full training set and require under an hour to train. For all 17 models (SAS and random subsets, full data) corresponding to a given dataset, we

[2] https://pytorch.org/tutorials/beginner/blitz/cifar10_tutorial.html

[3] https://github.com/sjoshi804/sas-data-efficient-contrastive-learning



(a) CIFAR10 (b) CIFAR100

Figure 1. CIFAR10 and CIFAR100 accuracies from the SAS paper

treated the number of epochs and all other hyperparameters as fixed. We felt this was reasonable as long as we were able to meaningfully compare test accuracies of models trained on SAS subsets, random subsets, and full data.

CIFAR100 and Food101 trained for 10 epochs each. CIFAR10, SVHN, and GTSRB trained for 5 epochs each, as these all reached high test accuracy relatively quickly. DTD trained for 40 epochs, as the training set was small and slow to improve accuracy. Due to the hardware constraints and time needed for training over 100 models, one difference between our approach and the SAS paper's is that we were only able to train and evaluate each model once as opposed to three times, so we expect somewhat more variability to be featured in our results.

5.3. Testing

Each model was evaluated on the full test set for its corresponding dataset, with results collected for the metrics outlined in Section 4.3, including overall (top-1) accuracy, per-class accuracy, and number of classes above zero accuracy. The most relevant metric to our discussion, in line with the original SAS paper's procedure, is top-1 accuracy.

6. Results and Analysis

In this section, we collect the results of our experiments with plots comparing top-1 test accuracy of SAS subsets, random subsets, and full data for the six datasets considered. With these representations, we seek to analyze the extent to which using the pre-trained ResNet CIFAR100 proxy model for subset selection on non-CIFAR100 datasets provides a benefit above random subset selection.

As a basis for comparison, in the original SAS paper, when the proxy model used in the upstream task was pretrained on the same data as the dataset under consideration, subsets selected by SAS generally outperformed those selected randomly by several percentage points at any given subset size (before converging on the accuracy of training on full data). Figure 1 depicts these results for CIFAR10 and CIFAR100.

6.1. CIFAR100 classification with CIFAR100 proxy model

We first decided to replicate the paper's task of selecting CIFAR100 SAS subsets with the CIFAR100 proxy model, so that this result could act as a point of comparison when selecting subsets of other data using the CIFAR100 proxy model. Though we were not able to achieve 66% test accuracy by training on the full train data set as in Figure 1 (ours was about 25%, limited by the model size, hardware, and time limitations), the same relative relationship between accuracies achieved by SAS, random, and full data still appeared to hold true, as seen in Figure 2. At any given subset size, SAS test accuracy generally outpaced that of the equivalent random subset by at least a few percentage points, roughly converging to the overall test accuracy of the full data model at around a subset size of 60-80%. This gave some support to the idea that our training procedure might yield results for the other datasets that could give a realistic picture of whether the CIFAR100 proxy model would be useful in selecting subsets for non-CIFAR100 data.

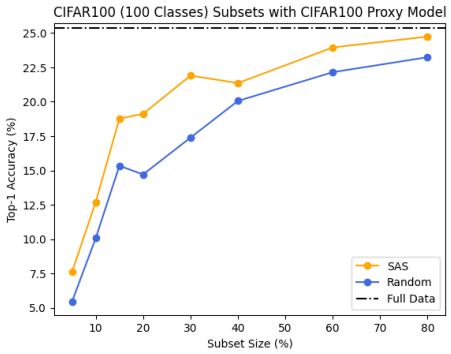


Figure 2. CIFAR100

6.2. Food101 classification with CIFAR100 proxy model

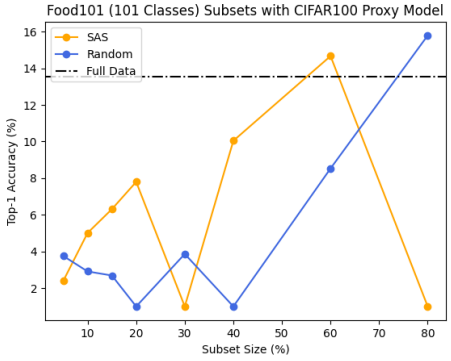


Figure 3. Food101

The Food101 dataset shares some surface level similarities to CIFAR100, with both datasets containing a roughly equal

number of class labels and balanced classes for the training and test sets, though the two datasets are comprised of entirely different data and sets of labels. This made Food101 a compelling choice to attempt SAS subset selection with the CIFAR100 proxy model.

The results from model evaluation as seen in Figure 3 seem to suggest an overall trend of SAS subsets outperforming random in this setting. This, however, comes with a few caveats, as (1) training accuracy on the full data only reached around 14% (though 95 of the 101 classes had above 0% test accuracy), and (2) at a handful of subset sizes, both SAS and random subsets failed to converge (the near-0 test accuracies in Figure 3).

Food101 is a large dataset that took the longest to train by far of the six we considered. The potential transferability of the CIFAR100 proxy model to a similar task is an exciting prospect but could be better validated with more training epochs on a larger neural network model specifically tailored to Food101, as well as by training each model multiple times and averaging the results.

6.3. CIFAR10 classification with CIFAR100 proxy model

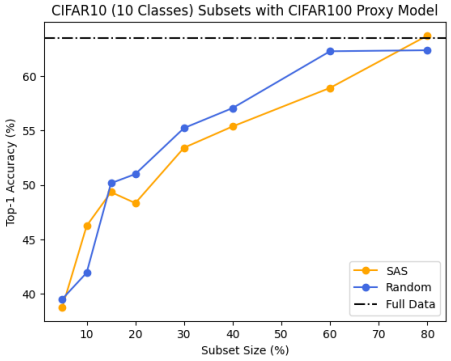


Figure 4. CIFAR10

CIFAR10 was another dataset of particular interest for performing SAS subset selection with the CIFAR100 proxy model, as these two datasets arguably contained the most similar data and classification tasks (though they do not share any class labels in common). Both sets contain 50,000 training points, 10,000 test points, and balanced classes.

However, unlike CIFAR100, where we saw better performance across all subsets generated through SAS, CIFAR10 subsets selected by SAS surprisingly returned test accuracies several percentage points lower than those returned by the random subsets, with the exception of the largest subset size of 80% (Figure 4). This result was in apparent contrast to the result in the original SAS paper, where CIFAR10 SAS subsets generated with a CIFAR10 proxy model consistently outperformed random. Our results also did not approach full data test accuracy until the subset contained about 80% of the training data, though that may have more to do with the specific classifier model we trained.

In light of this result, it seems possible that the proxy model, in attempting to estimate expected augmentation similarities between training points, may have resulted in a somewhat skewed and incorrect "understanding" of what constitutes an appropriate cluster center, resulting in slightly less classification abililty than simply selecting data points randomly.

6.4. SVHN classification with CIFAR100 proxy model

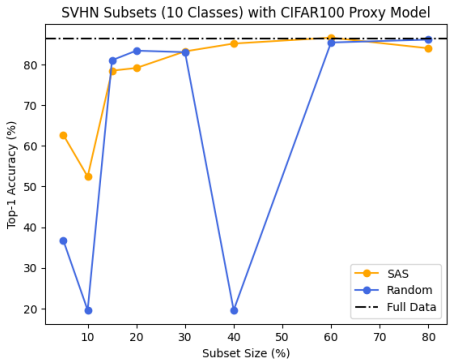


Figure 5. SVHN

We included the SVHN dataset in order to evaluate a 10-label classification task with data that was more dissimilar to CIFAR100 than CIFAR10. CIFAR100 and CIFAR10 both involve classifying animals and objects, whereas SVHN involves correctly identifying digits. Also, SVHN contains imbalanced classes (digits 1 and 2 appear 2-3 times as often as some other digits), unlike CIFAR100 and CIFAR10. Our reasoning was that, in theory, if the CIFAR100 proxy model provided any benefit to selecting subsets for CIFAR10 through SAS, that effect might be absent for SVHN.

The results for this dataset were somewhat ambiguous. As seen in Figure 5, better performance can be seen for the subsets 5%, 10%, 30%, 40%, and 60% when utilizing SAS, while the remaining sizes 15%, 20%, and 80% perform at a higher test accuracy when using randomly generated subsets. At convergence we were able to achieve 86% test accuracy, though some random subset models failed to converge. Additionally, both random and SAS subsets converged on full data test accuracy with about 30% of the the data (similar to what was observed for CIFAR10 in the SAS paper, but not in our results for CIFAR10), indicating subsets at or below 30% are the most meaningful points to consider. Retraining the models multiple times and averaging their evaluation results would confirm this, but one possible takeaway is that, above the smallest subset sizes of 5% and 10% and below 30%, random selection seems to outperform SAS.

6.5. GTSRB classification with CIFAR100 proxy model

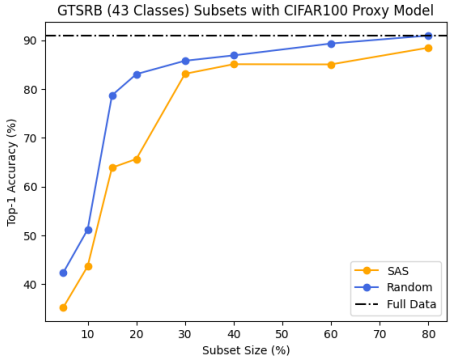


Figure 6. GTSRB

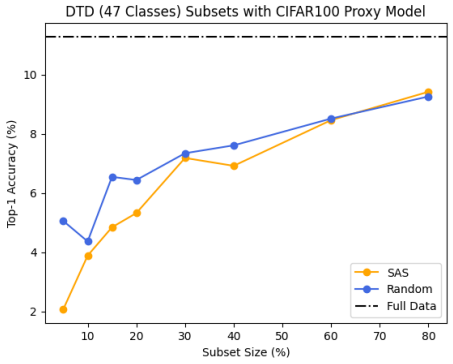


Figure 7. DTD

In this and the following section, we respectively examined the GTSRB (43 classes) and DTD (47 classes) datasets to test the usability of the CIFAR100 proxy model for subset selection between the relative extremes of the 100-class and 10-class prediction tasks. GTSRB involves classifying road signs, a similar object identification task to the other datasets discussed, though of course containing different types of images to CIFAR100. GTSRB also features a degree of class imbalance, with some of the most frequent classes appearing over 10x more frequently than the least frequent.

For this dataset, we were able to achieve a high performance of above 90% test accuracy when training on full data. We also observed consistent results across all subset sizes, with random subset test accuracy always outperforming SAS subset accuracy by at least a few percentage points (over 15 percentage points at some subset sizes), per Figure 6. These results seemed to reasonably suggest that SAS subset selection with the CIFAR100 proxy model was not an appropriate choice in this setting and most likely resulted in a skewed augmentation similarity function that yielded non-representative "class centers".

6.6. DTD classification with CIFAR100 proxy model

Finally, we included DTD as another dataset with a "medium" number of classes because it represented a challenging classification task, describing visual textures displayed in images (though with balanced classes). We felt this task would differ fairly substantially from CIFAR100 and most likely result in SAS subsets performing worse than random. That appears to be the result per Figure 7, where we observed that randomly generated subsets tested at a higher accuracy for all subsets sizes except 80%, where performance was nearly identical to SAS. These results do come with similar caveats to other datasets like Food101, as the overall test accuracy we were able to achieve was relatively low at roughly 11% (we believe it was because this classification task was particularly difficult). Even for 80% subset sizes, SAS and random both failed to approach the test accuracy of the full data model. However, for the results we did see, the relatively superior performance of random subsets over SAS was notable.

7. Conclusion

In conducting our testing, we recognized that more robust results could have been attained by training larger models on GPU for more epochs, customizing the model design used per dataset, and retraining models multiple times so that results could be averaged. However, we do believe that under our constraints, we did observe interesting results that could be validated with further testing.

As expected, subsets selected by SAS using the pre-trained CIFAR100 proxy model performed best when CIFAR100 was the dataset being tested, as this model would be the most representative during the key step of estimating augmentation similarity when selecting subsets. However, when extending use of the CIFAR100 proxy model to classification tasks CIFAR10, SVHN, GTSRB, and DTD, where the number of labels was much smaller than 100, it seemed that the CIFAR100 proxy model might not be an appropriate choice for SAS. For these datasets, SAS subsets of a given size generally performed worse than random, and this result appeared to not innately depend on class balance, as SAS underperformed even for balanced datasets like CIFAR10 and DTD.

Testing on the Food101 dataset yielded potentially interesting results, with the general trend seeming to show SAS subsets outperforming random. If these results are representative, this may be because Food101 and CIFAR100 share a similar number of classes and might lead to further research questions, such as whether a proxy model would transfer better to a dissimilar dataset with a similar number of classes (CIFAR100 vs. Food101) than to a similar dataset with a different number of classes (CIFAR100 vs. CIFAR10).

8. Contributions

- Matt Abuzalaf - Dataset Research, SAS Subset Collection, Classifier Model Design/Training/Testing/Plots, Report contributions on Formulation/Method/Experiments/Results/ Conclusion
- Ryan Dunker - Dataset Research, SAS Subset Collection, Report contributions on Introduction/Results
- Yeonje Jo - SAS Subset Collection, Application of PyTorch tutorial to train and test SAS and compare with random subsets, Report contributions on Abstract/Introduction/Method/Experiments
- Lauren Mizner - SAS Subset Collection, Application of PyTorch tutorial to train and test SAS and compare with random subsets, Report contributions on Related Work/Problem Formulation/Results/Conclusion

References

Bossard, L., Guillaumin, M., and Van Gool, L. Food-101 – mining discriminative components with random forests. In *European Conference on Computer Vision*, 2014.

Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., , and Vedaldi, A. Describing textures in the wild. 2014.

Houben, S., Stallkamp, J., Salmen, J., Schlipsing, M., and Igel, C. Detection of traffic signs in real-world images: The German Traffic Sign Detection Benchmark. In *International Joint Conference on Neural Networks*, number 1288, 2013.

Joshi, S. and Mirzasoleiman, B. Data-efficient contrastive self-supervised learning: Most beneficial examples for supervised learning contribute the least. 2023. URL https://arxiv.org/pdf/2302.09195.pdf.

Krizhevsky, A. Cifar10. URL https://www.cs.toronto.edu/~kriz/cifar.html.

Krizhevsky, A., Nair, V., and Hinton, G. Cifar-100 (canadian institute for advanced research). URL http://www.cs.toronto.edu/~kriz/cifar.html.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. Learning transferable visual models from natural language supervision. 2021. URL https://arxiv.org/pdf/2103.00020.pdf.

Yuval Netzer, Tao Wang, A. C. A. B. B. W. A. Y. N. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*. URL http://ufldl.stanford.edu/housenumbers.

A. Code

https://github.com/mabu1996/CS260_Final_Project/tree/main

B. Accuracy tables

The tables below display the top-1 accuracies for classifier models trained on subsets for each of the six datasets used for this project. Additional metrics were gathered for this project but did not fit well in a LaTeX view but can be seen in the output of our project notebook at the Github link above. These metrics include: per-class accuracy, worst-class accuracy, best-class accuracy, number of classes above 0 accuracy.

Subset size	Random	SAS
0.05	0.0545	0.0761
0.10	0.1007	0.1270
0.15	0.1534	0.1877
0.20	0.1471	0.1912
0.30	0.1739	0.2190
0.40	0.2006	0.2136
0.60	0.2215	0.2395
0.80	0.2324	0.2474
Full data	0.2537	0.2537

Table 2. Test accuracy for different subset sizes and selection methods of CIFAR100

Subset size	Random	SAS
0.05	0.037465	0.023921
0.10	0.029030	0.050059
0.15	0.026812	0.063050
0.20	0.009901	0.077941
0.30	0.038574	0.009901
0.40	0.009901	0.100396
0.60	0.085267	0.146653
0.80	0.157901	0.009901
Full data	0.135327	0.135327

Table 3. Test accuracy for different subset sizes and selection methods of Food101

Subset size	Random	SAS
0.05	0.3952	0.3875
0.10	0.4193	0.4624
0.15	0.5016	0.4934
0.20	0.5100	0.4833
0.30	0.5523	0.5342
0.40	0.5707	0.5538
0.60	0.6229	0.5892
0.80	0.6238	0.6373
Full data	0.6348	0.6348

Table 4. Test accuracy for different subset sizes and selection methods of CIFAR10

Subset size	Random	SAS
0.05	0.367432	0.626805
0.10	0.195874	0.524239
0.15	0.810426	0.784726
0.20	0.833897	0.791295
0.30	0.830171	0.832091
0.40	0.195874	0.851375
0.60	0.853949	0.865819
0.80	0.861094	0.839774
Full data	0.863245	0.863245

Table 5. Test accuracy for different subset sizes and selection methods of SVHN

Subset size	Random	SAS
0.05	0.423674	0.352415
0.10	0.511876	0.436580
0.15	0.787015	0.638717
0.20	0.830404	0.656453
0.30	0.857799	0.831037
0.40	0.868804	0.850752
0.60	0.893112	0.850356
0.80	0.909422	0.884561
Full data	0.910055	0.910055

Table 6. Test accuracy for different subset sizes and selection methods of GTSRB

Subset size	Random	SAS
0.05	0.050532	0.020745
0.10	0.043617	0.038830
0.15	0.065426	0.048404
0.20	0.064362	0.053191
0.30	0.073404	0.071809
0.40	0.076064	0.069149
0.60	0.085106	0.084574
0.80	0.092553	0.094149
Full data	0.112766	0.112766

Table 7. Test accuracy for different subset sizes and selection methods of DTD